iTOUGH2

Short Course

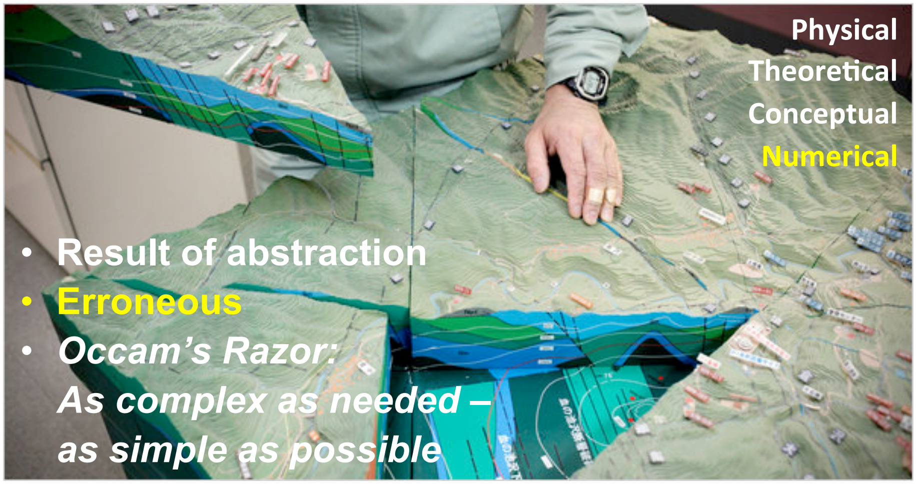
Stefan Finsterle

Lawrence Berkeley National Laboratory
Earth Sciences Division
Berkeley, California

Outline

- About Modeling
- Forward modeling and TOUGH2
- Inverse Modeling and iTOUGH2
- Course Objectives

Subsurface models: then and now



"A model of the Yanaizu-Nishiyama geothermal plant. Japan's 18 geothermal plants account for only 0.3 percent of its electricity production."

A. Pollack, Japan's Nuclear Future in the Balance, *New York Times*, May 9, 2011.

A model is…

"… a *purposeful*, *simplified* representation of a real system"

"… a simple worldview with an attitude"

"*simple and with purpose*":

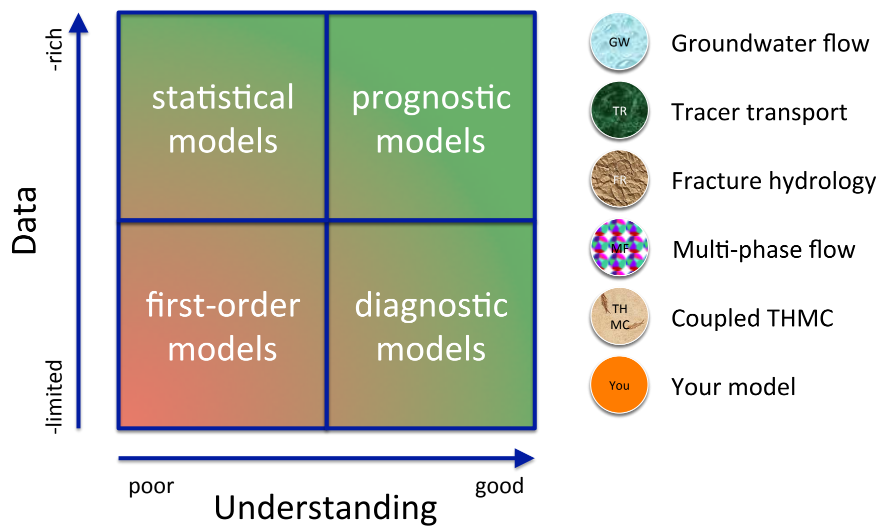
The model is *simple* in that it contains only features of *primary importance* for the *intended use* of the model

Occam's razor

"Non sunt multiplicanda entia praeter necessitatem"

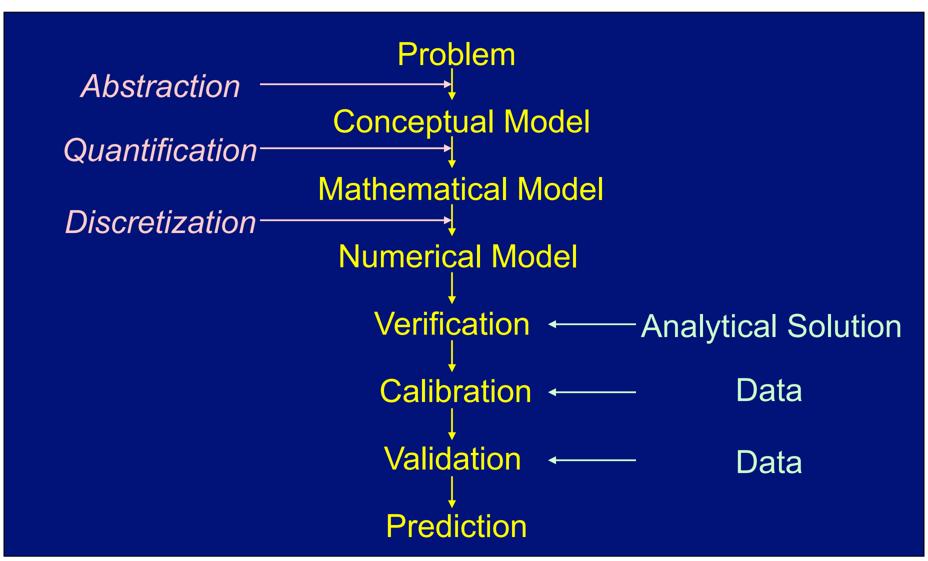
Make it as complex as needed, …
but *keep it as simple as possible!*

System understanding, data availability, and role of modeling



after Holling (1978)

Model Development



6

Role of Mathematical Modeling

Improve process understanding TOUGH2
- Understand nonlinearities/coupled processes
- Evaluate non-observable quantities
- What-if scenarios ("virtual sandbox")

iTOUGH2

Design experiments
- Identify experimental procedure yielding data that contain information about relevant properties

Analyze data
- Determine parameters from data
- Identify model structure

Decision support
- Risk assessment
- Sensitivity analysis

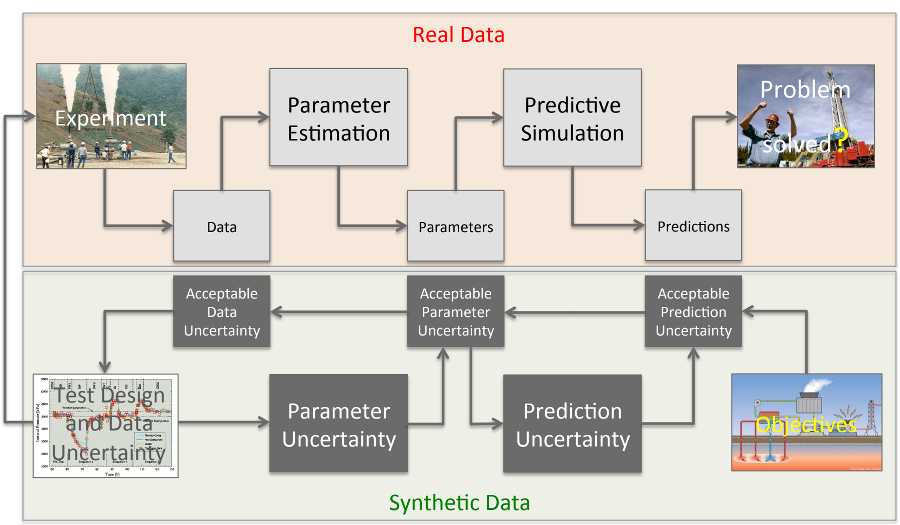
Make predictions
- Deterministic/probabilistic
- Sensitivity analysis

Uncertainty quantification

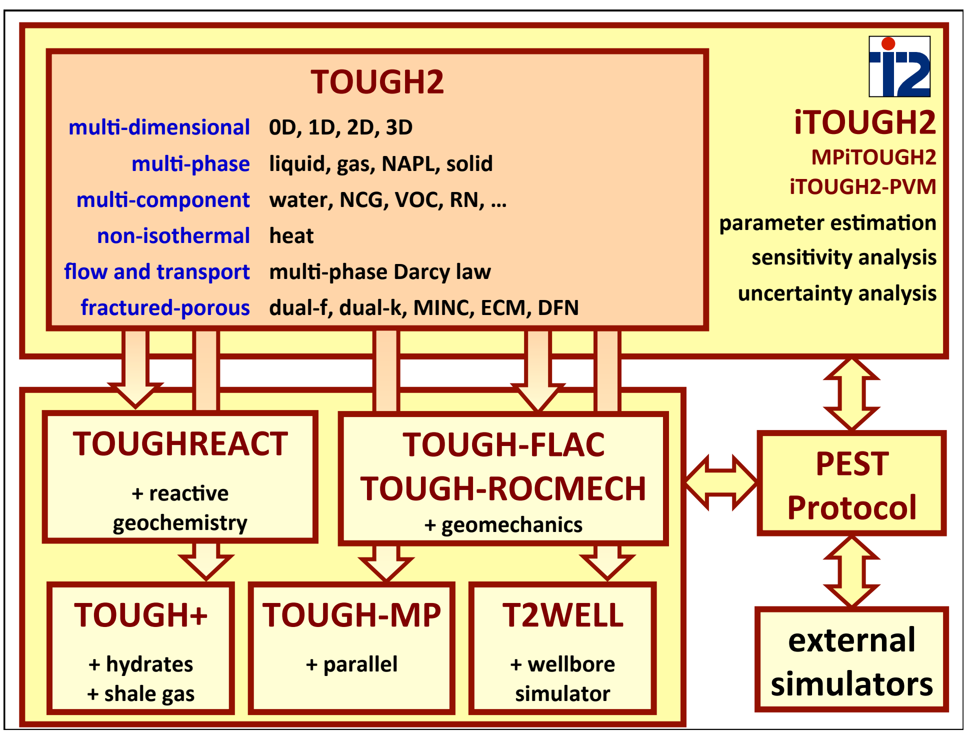
Optimization

7

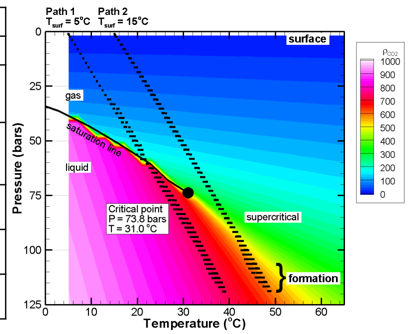
Uncertainty drives your modeling,…

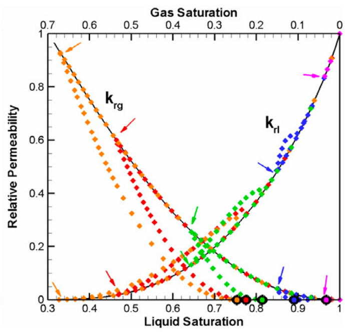


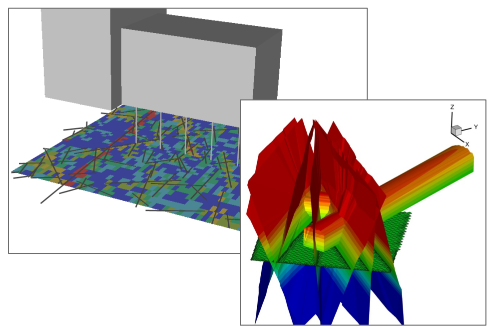

TOUGH
Suite of
Nonisothermal Multiphase Flow
and Transport Simulators


TOUGH2
multi-dimensional
0D, 1D, 2D, 3D
multi-phase
liquid, gas, NAPL, solid
multi-component
water, NCG, VOC, RN, …
non-isothermal
heat
flow and transport
multi-phase Darcy law
fractured-porous
dual-f, dual-k, MINC, ECM, DFN
iTOUGH2
MPiTOUGH2
iTOUGH2-PVM
parameter estimation
sensitivity analysis
uncertainty analysis
TOUGHREACT
+ reactive geochemistry
TOUGH-FLAC
TOUGH-ROCMECH
+ geomechanics
PEST Protocol
TOUGH+
+ hydrates
+ shale gas
TOUGH-MP
+ parallel
T2WELL
+ wellbore simulator
external simulators

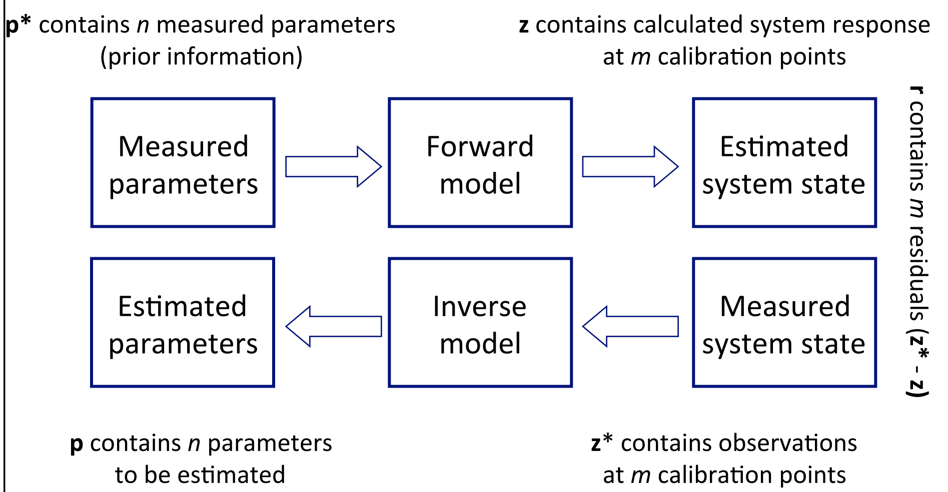
Components κ	Phase β			
	Solid	Fluid		
		Liquid		Gas
		Aqueous	NAPL	
Water	Ice	Liquid water	Water dissolved in NAPL	Water vapor
NCG	-	NCG dissolved in liquid water	NCG dissolved in NAPL	Non-condensible gas
VOC	-	VOC dissolved in liquid water	VOC in NAPL phase	Chem. vapor
Minerals	Grains	Dissolved, Colloids	Dissolved, Colloids	-







Forward Modeling



13

- Mathematical model G relates parameters **p** to observations **z** $G(\mathbf{p}) = \mathbf{z}$
- Observations have noise:

$$\mathbf{z} = G(\mathbf{p}) + \varepsilon = \mathbf{z}_{true} + \varepsilon$$

- Forward problem: find **z** given **p**
- Inverse problem: find **p** given **z**

$$\mathbf{p} = G^{-g}(\mathbf{z})$$

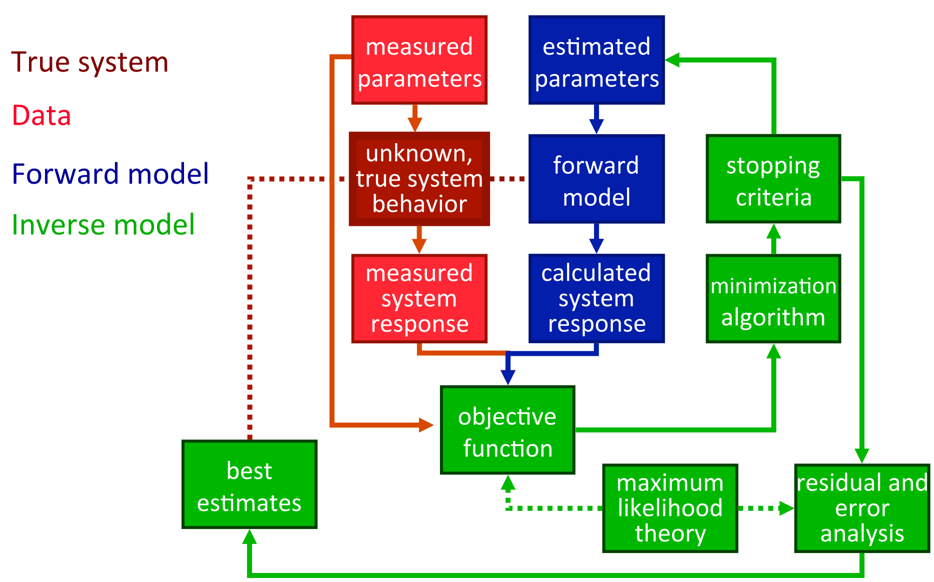
- Model identification problem: find G given examples of **p** and **z**
- Discrete inverse problem: **z** is a vector of observations at discrete points in space and time (“calibration points”)

14

- Inverse modeling ≈ parameter estimation ≈ model calibration ≈ history matching ≈ curve fitting ≈ regression (≈ optimization ≈ filtering)
- distinction is largely irrelevant
- preferred terms in different disciplines:
  - inverse problem: generally many parameters; ill-posed; geophysics; imaging; applied mathematics
  - parameter estimation: generally few parameters; well-posed; hydrogeology
  - model calibration: process modeling
  - history matching: reservoir engineering
  - curve fitting: data analysis
  - regression: statistics

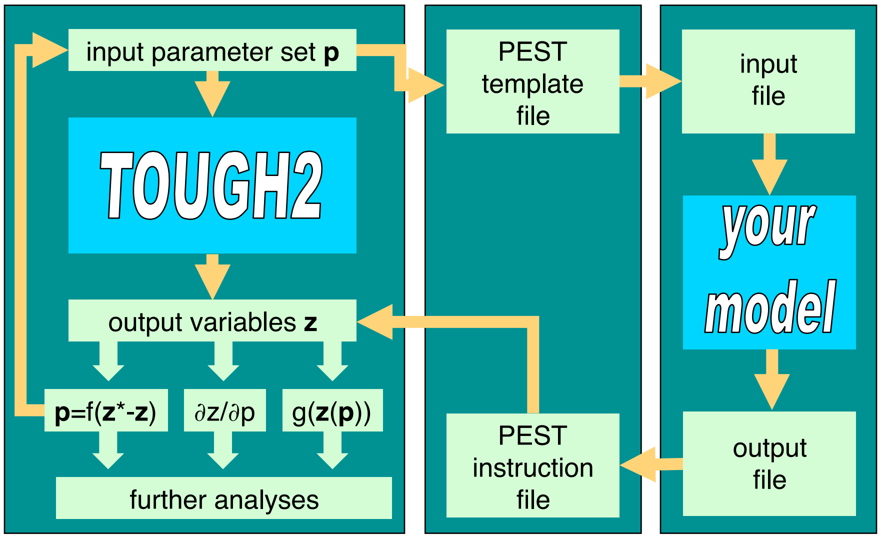
15

iTOUGH2 flowchart



16

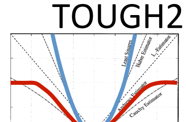
Running TOUGH2 once is tough; running it many times is iTOUGH2

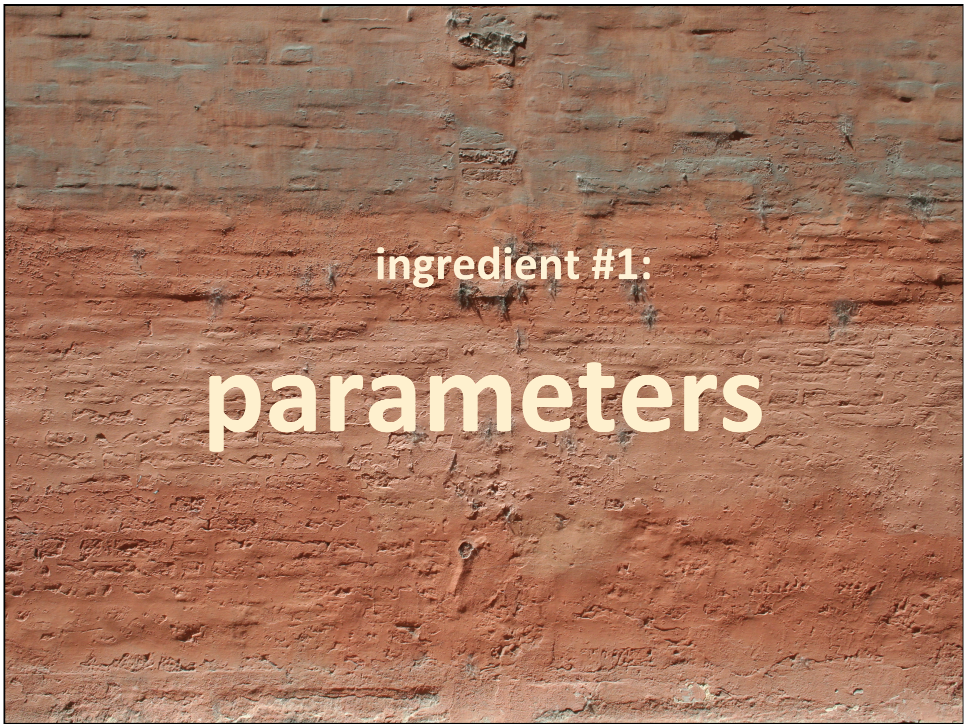


17

Inverse modeling ingredients

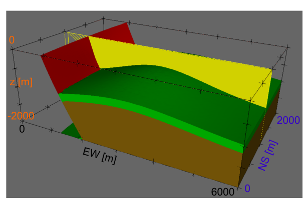
- Parameters — **p** — Parameterization
- Data — **z*** — Joint inversion
- Forward operator — **z**(**p**) — TOUGH2
- Objective function — *S*(**z***-**z**(**p**))
- Minimization algorithm — min *S*(**p**)
- Sanity check — S_{min}, σ_p, σ_z — *uncertainty analysis*

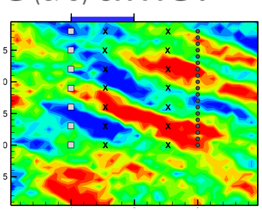




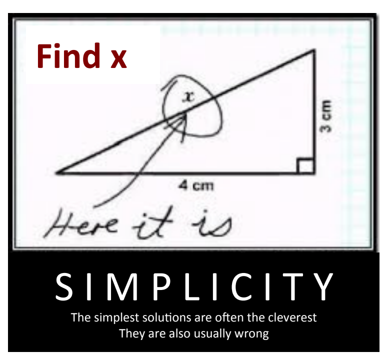
So many parameters…so little (CPU) time!

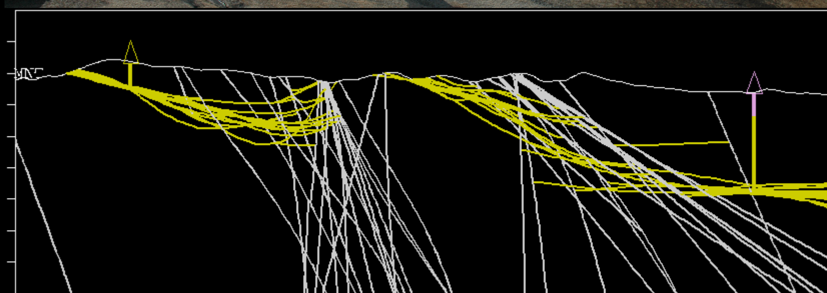
- Parameterization
  - Properties
  - Structure
  - Forcing terms
  - Conceptual errors
  - Joint estimation of properties, structure, forcing terms, errors





- Occam's razor
  - *as complex as needed – as simple as possible*
  - Start simple and add complexity
  - Start complex and simplify
- Parameter selection
  - Sensitivity
  - Independence
  - Superparameters

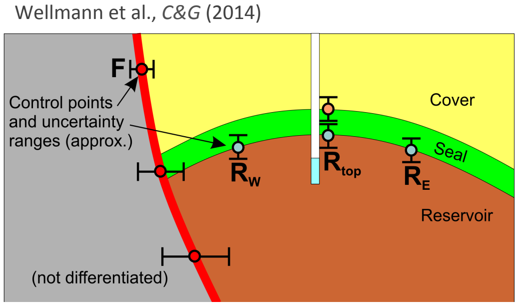


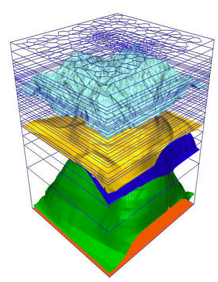


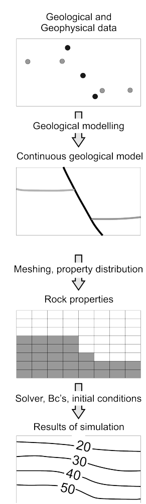
Different models created by different geologists for the same region (Courrioux et al., Experience from 3D cartography training sessions. Implication for uncertainty factors identification. 34th International Geological Congress, August 5-10, 2012, Brisbane, Australia)

Joint analysis of structural and property data reduces estimation and prediction bias

Wellmann et al., *C&G* (2014)







- Parameterize geologic structure at control points
- Develop integrated workflow from structural geologic data to discretized reservoir model
- Jointly estimate structural geometry and reservoir properties

Want more parameters?
Give me more data!

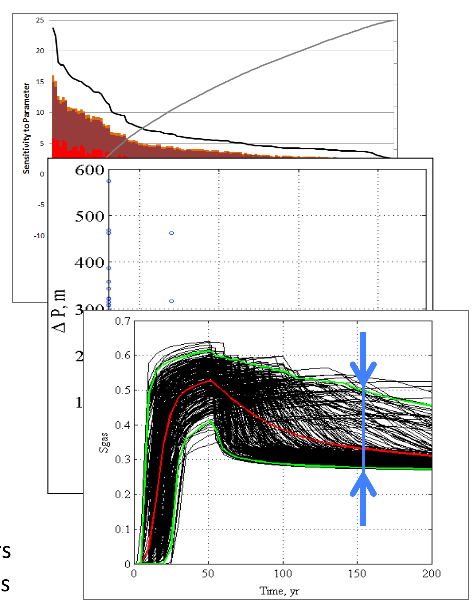
The more
high-quality,
sensitive,
complementary,
consistent
data, the better

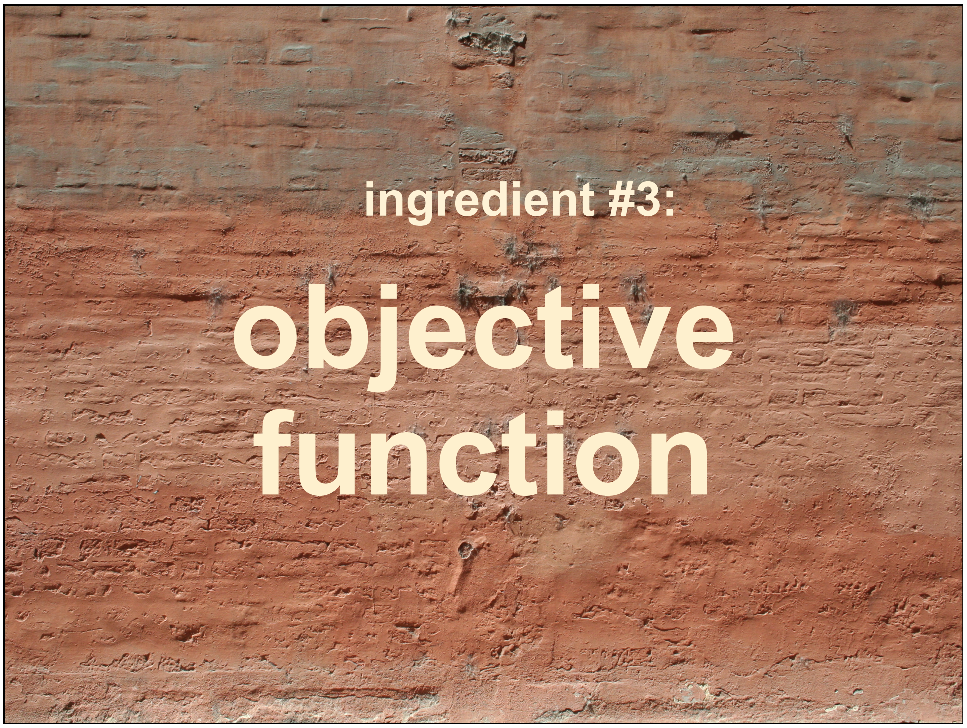
mixing ingredients #1 and #2:

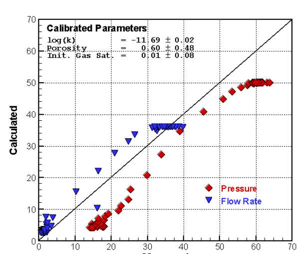
sensitivity analysis

Going global with sensitivity analysis

- Local sensitivity analysis
  - Requires (only) n+1 (parallel) runs
  - By-product of derivative-based minimization algorithms
  - Many useful composite measures on parameter influence and data worth
- Morris One-At-A-Time
  - Requires $r(n+1)$ runs
  - Identifies influential and non-influential parameters
  - Identifies nonlinearity and interaction effects
- Sobol'/total sensitivity indices
  - variance/sampling based
  - Identifies contribution of uncertain parameter to prediction uncertainty
  - Fix one parameter – vary all the others
  - Vary one parameter – fix all the others







heteroscedastic

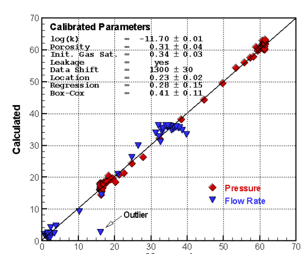
$$\tilde{z}(z;\lambda) = \begin{cases} \dfrac{(z^{\lambda}-1)}{\lambda \cdot g^{\lambda-1}} & \text{if } \lambda \neq 0 \\ g \cdot \ln(z) & \text{if } \lambda = 0 \end{cases}$$

autocorrelated

$$\tilde{r}_i = \begin{cases} r_1\sqrt{(1-\rho^2)} & \text{if } i = 1 \\ r_i - \rho \cdot r_{i-1} & \text{if } i = 2,\ldots,m \end{cases}$$

skewed and heavy-tailed

$$p(r_i \mid \xi, \beta) = \frac{2\sigma_{\xi}}{\xi + \xi^{-1}}\omega_{\beta}\exp\left\{-c_{\beta}\left|r_{\xi,i}\right|^{2/(1+\beta)}\right\}$$



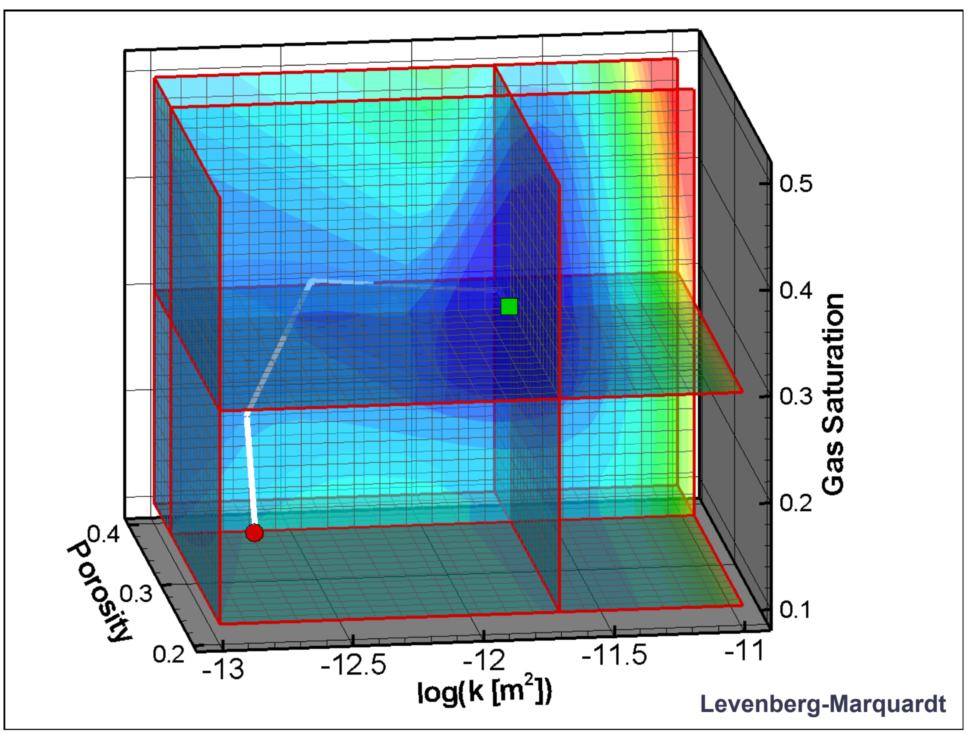
concurrently estimate *statistical* parameters
$\beta, \lambda, \rho, \xi$

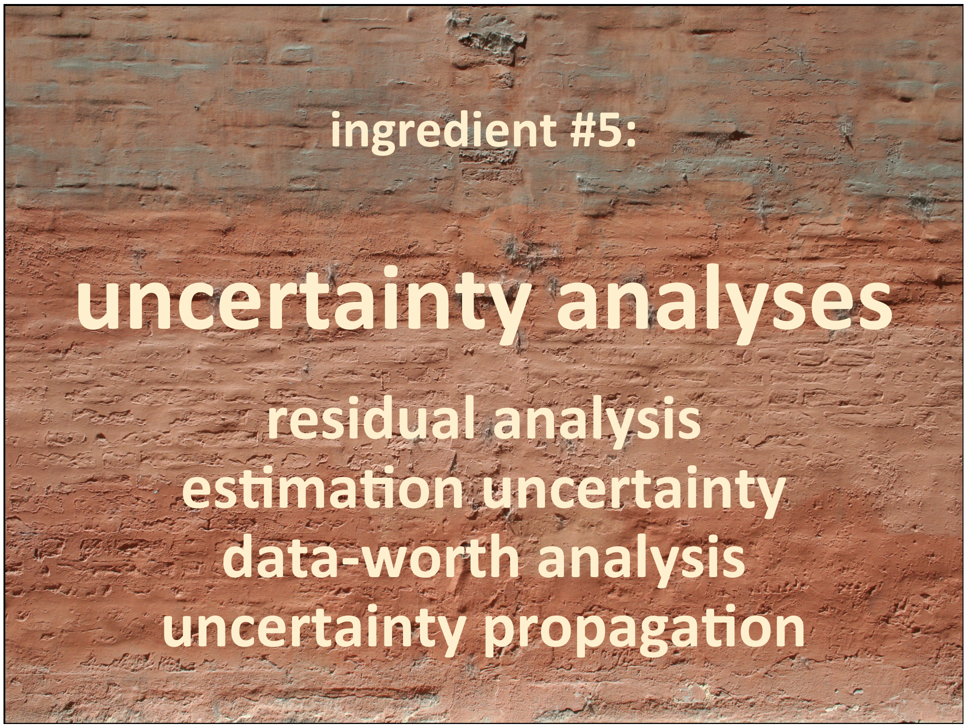
Schoups and Vrugt, *WRR*, 2010; Finsterle and Zhang, *C&G*, 2011


ingredient #4:
minimization
algorithm


Derivative-Based Minimization Algorithms
iTOUGH2
iTOUGH2
iTOUGH2
iTOUGH2


Harmony Search
Swarm Optimization
Evolutionary Algorithms
Simulated Annealing


Gas Saturation
0.5
0.4
0.3
0.2
0.1
Porosity
0.4
0.3
0.2
-13
-12.5
-12
-11.5
-11
log(k [m²])
Levenberg-Marquardt

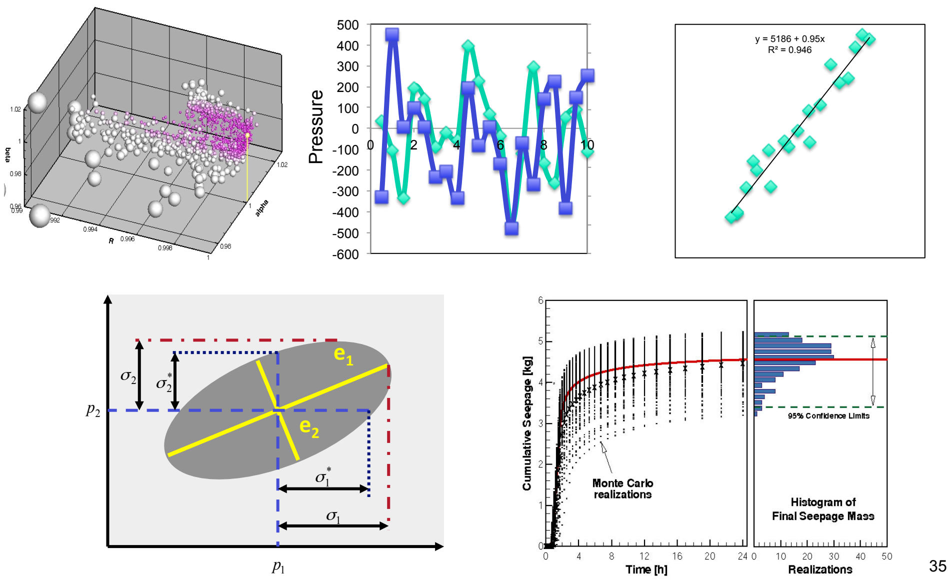


Why residual and uncertainty analysis?

- Parameter estimates may be worthless if:
  - Model does not match the data, i.e., is an unlikely representation of the true system
    goodness-of-fit, Fisher Model Test
  - Estimates are biased by systematic errors or outliers in the data
    residual analysis
  - Estimation uncertainty is large
    C_{pp}, *correlation coefficients*
  - Solution is *non-unique* or *unstable*
  - Predictions are highly *uncertain*

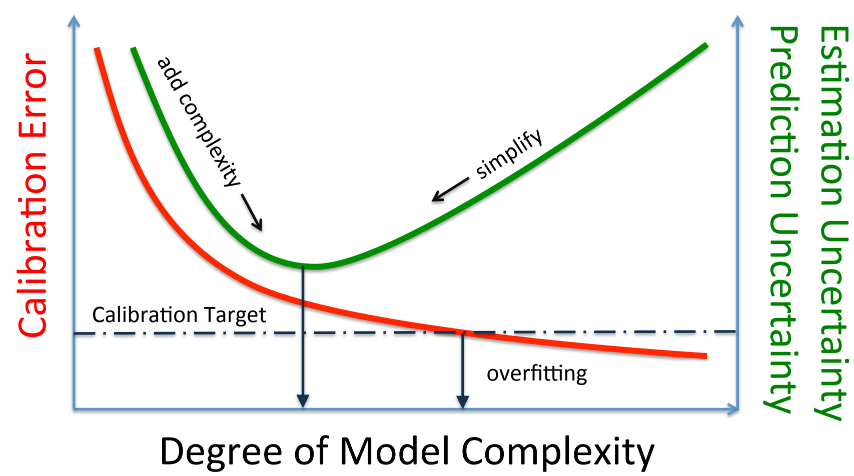
34

Residual and Uncertainty Analysis



35

Occam's Razor - sharpened!



Success Criteria

- Captures salient features of system behavior
  (expert judgment)
- Acceptable match
  (goodness-of-fit criteria)
- Acceptable estimation uncertainty
  (determinant of estimation covariance matrix)
- Ability to make acceptable predictions
  (validation acceptance criteria)
- Combination
  (model identification criteria)

➢ Depends on study objectives

➢ Use as criteria for test design!

37

Textbooks

- Aster et al., *Parameter Estimation and Inverse Problems*, 2nd, Ed., Academic Press, 2013.
- Hill and Tiedeman, *Effective Groundwater Model Calibration, With Analysis of Data, Sensitivities, Predictions, and Uncertainty*, Wiley, 2007.
- Saltelli et al., *Global Sensitivity Analysis, The Primer*, Wiley, 2008.
- …

38

Course Objectives

- **General:** Provide participants with *conceptual understanding*, *theoretical background*, and *practical experience* in solving simulation-*optimization* problems in geosciences and engineering using *iTOUGH2*.
- **Lectures:** Understand optimization and uncertainty quantification:
  - Fundamental concepts
  - Theoretical basis
  - Practical approaches
  - Interactive discussions
- **Computer Exercises:** Gain experience using iTOUGH2
- **Course Project:** Define and develop a simulation-optimization problem of interest to you

40

iTOUGH2 Short Course

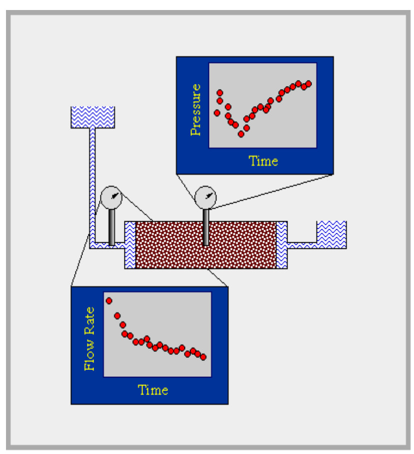
Lawrence Berkeley National Laboratory
Earth Sciences Division
Berkeley, California

Computer Exercises: Overview

Darcy
Polynomial
CO_2 Injection Test Design
Five-Spot Geothermal Injection-Production
Standard Test Set
Project

Tutorial Problem: Darcy

- Objectives:
  - Understand main iTOUGH2 concepts
  - Get familiar with key iTOUGH2 input blocks
  - Get familiar with iTOUGH2 output file
  - Examines impact of measurement noise on estimates
  - Requires some knowledge of TOUGH2 simulator
- Parameter Estimation Problem:
  - Estimate 3 parameters…
  - …based on transient pressure and flow-rate data…
  - …using Levenberg-Marquardt minimization algorithm



42

Variations of Darcy Problem

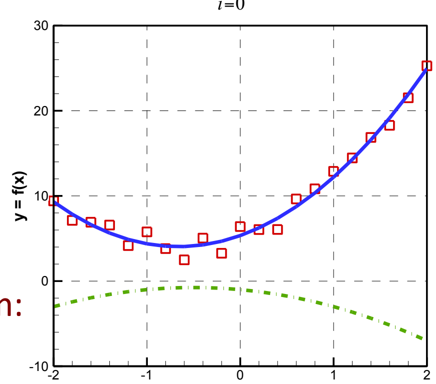
- darcy1i: run forward problem
- darcy2i: inversion with perfect data (*nonoise.dat*)
- darcy3i: inversion with noisy data (*noisy.dat*)
- darcy4i: explore
- darcy5i: grid search
- darcy6i: Morris global sensitivity analysis
- darcy7i: Saltelli/Sobol' global sensitivity analysis
- darcy8i: Monte Carlo (LHS) uncertainty quantification

43

iTOUGH2-PEST Tutorial Problem: Polynomial

- Objectives:
  - Understand main iTOUGH2 concepts
  - *Get familiar with PEST protocol*
  - Get familiar with key iTOUGH2 input blocks
  - Get familiar with iTOUGH2 output file
- Parameter Estimation Problem:
  - Estimate coefficients of polynomial…
  - …using Levenberg-Marquardt minimization algorithm

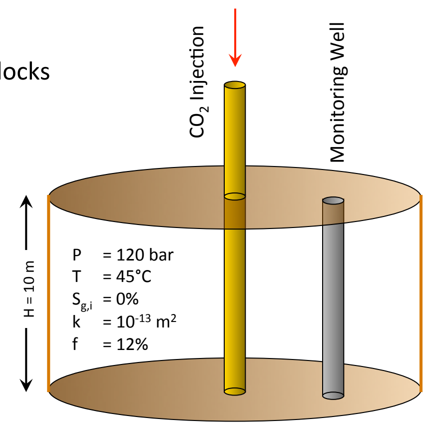
$$y(x) = \sum_{i=0}^{n} a_i \cdot x^i$$



44

Tutorial Problem: CO_2 Injection

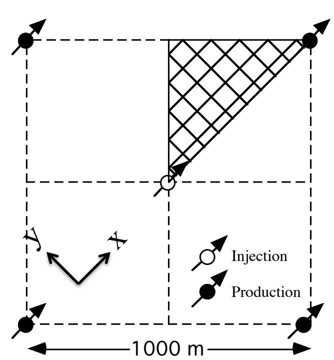
- Objectives:
  - Understand main iTOUGH2 concepts
  - Get familiar with key iTOUGH2 input blocks
  - Get familiar with iTOUGH2 output file
  - Requires some knowledge of TOUGH2 simulator (module ECO2N)
- Analyses performed:
  - Forward simulation
  - Sensitivity analysis
  - Data inversion
  - Uncertainty propagation analysis



45

Five-Spot Geothermal Injection-Production Problem

- Objectives:
  - Understand main iTOUGH2 concepts
  - Get familiar with key iTOUGH2 input blocks/ commands and output files
  - Requires some knowledge of TOUGH2 simulator (module EOS1)
- Exercises:
  1. TOUGH2 simulation with iTOUGH2
  2. Generation of synthetic data
  3. Defining parameters and performing sensitivity analysis
  4. Inversion of synthetic data
  5. Uncertainty propagation analysis
  6. Explore



46

Standard iTOUGH2 Samples and Installation Test Cases

- sample1-7: see report *iTOUGH2 Sample Problems*
- sampleLOCAL: local minimization algorithms
- sampleGLOBAL: global minimization algorithms
- sampleGS: user-specified sets (eos7c)
- sampleGSLIB: geostatistics
- sampleMOAT: Morris global sensitivity analysis
- samplePARALLEL: parallel execution
- samplePARETO: Pareto frontier using iTOUGH2-PEST
- sampleREGION: Permeability region

47

Formulate Optimization Problems

Parameters	Observations	Objective Function
Problem A:		
Problem B:		

49

iTOUGH2 Short Course

Lawrence Berkeley National Laboratory
Earth Sciences Division
Berkeley, California

iTOUGH2 Tutorial

iTOUGH2 Concept
Input Language
Running iTOUGH2

Forward model definition

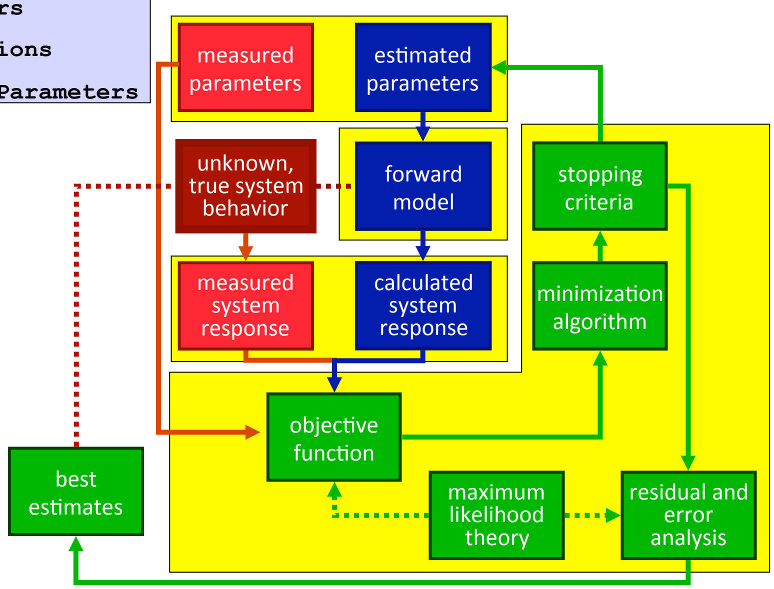
Inverse model definition

`> Parameters`

`> Observations`

`> Control Parameters`

TOUGH2 Model



2

Elements of iTOUGH2 Input Language

- Text-based; structured command input language
- Command Level Markers

>, >>, >>>, >>>>
<, <<, <<<, <<<<

Commands and optional **keywords**

```
>> TIME in MINUTES
```

: Parameter Value

```
>>> ELEMENT: ELM_1 ELM_2
>>>> VARIANCE: 10.0
```

Commands and keywords are case-*insensitive*

3

First-Level Commands

> PARAMETER

identifies parameters to be estimated

(refers to *input* to forward model)

> OBSERVATION

defines calibration points in space and time and reads related measured data

(refers to *output* from forward model)

> COMPUTATION

program options, computational parameters

4

> PARAMETERS

- Select parameters to be estimated or analyzed
- These parameters are a (potentially transformed) subset of *input* variables to the forward model
- The selected parameters are the *n* entries of the parameter vector **p**

```
> PARAMETERS
  >> specify parameter type
     >>> specify parameter domain
         >>>> provide details
         <<<<
     <<<
  <<
```

5

> PARAMETER Example

```
> PARAMETER
  >> ABSOLUTE permeability
     >>> MATERIAL: SAND1
         >>>> LOGARITHM
         >>>> RANGE: -14.0 -10.0
         >>>> INDEX: 3 (vertical)
         >>>> initial GUESS: -12.0
         >>>> standard DEVIATION: 0.5
         <<<<
```

6

> OBSERVATIONS

- Select observations used to estimate parameters
- These observations refer to *output* variables from the forward model and to the corresponding measured data
- The selected observations are the *m* entries of the observation vectors **z** (calculated) and **z*** (measured)

```
> OBSERVATION
  >> specify calibration TIMES
  >> specify observation type
     >>> specify location
         >>>> provide details
         >>>> provide DATA
         <<<<
     <<<
  <<
```

7

> OBSERVATION Example

```
> OBSERVATION
  >> TIMES: 100 LOGARITHMIC MINUTES
     1.0 1440.0
  >> GAS FLOW RATE
     >>> CONNECTION: ABC99 OUT99
         >>>> FACTOR: -1.0
         >>>> RELATIVE: 10%
         >>>> DATA on FILE: qgas.dat
         <<<<
```

8

> COMPUTATION

```
> COMPUTATION
  >> program OPTIONS
  >> STOPPING criteria
  >> JACOBIAN
  >> ERROR analysis
  >> OUTPUT options
  <<
```

9

> COMPUTATION
Example

```
> COMPUTATION
  >> program OPTIONS
     >>> LEVENBERG-MARQUARDT
     >>> PEST
     <<<
  >> STOPPING criteria
     >>> ITERATIONS: 10
     >>> ignore WARNINGS
     >>> STEP: 5.0
     <<<
  >> JACOBIAN
     >>> CENTERED
     >>> PERTURB: 1 %
     <<<
  >> ERROR analysis
     >>> ALPHA: 5 %
     >>> MONTE CARLO
     <<<<
  >> OUTPUT options
     >>> OBJECTIVE FUNCTION
     >>> SENSITIVITY matrix
     >>> NEW OUTPUT
     >>> FORMAT: COLUMN
     >>> MINUTES
     <<<
  <<
```

10

Commenting

- Lines without a command level marker (> or <) are considered comments.
- Any text other than commands or keywords acceptable as comment.
- Commenting out a single line:

  `#` in first column
- Commenting out multiple lines (block):

  `/*` Beginning of block

  `*/` End of block
- `INCLUDE FILE: file_name`
- `ECHO ON/OFF` (echoes commands to msg file)

11

Running iTOUGH2 (Unix)

`itough2 it2_file_name t2_file_name IEOS &`

itough2	it2_file_name	t2_file_name	IEOS
command	iTOUGH2 input file (inverse problem)	TOUGH2 input file (forward problem)	EOS module

Examples:

`itough2` (command usage and options)

```
itough2 darcyi darcy 3 &
itough2 -mesh -i test.inc testi test 9 &
itough2 -pest it2_control_file &
```

12

Running iTOUGH2 (PC)

- Copy appropriate iTOUGH2 executable (e.g., `iT2_3.exe` or `iT2_PEST.exe`) from directory *Executable* to working directory and double-click on it
- Type name of iTOUGH2 input file (e.g., `darcy1i`)
- Type name of TOUGH2 input file (e.g., `darcy`; no forward file needed if iT2_PEST is used)
- To run a forward problem only, provide a dummy iTOUGH2 input file (e.g., `invdir`, or an empty file)
- You may install batch files (`tough2.bat` and `itough2.bat`) to conveniently run iTOUGH2 from a DOS Command Prompt using Unix syntax.

13

Running iTOUGH2 on PC using Batch File

- Locate directory **Executable**, which contains the *itough2.bat* batch file and executables *it2_EOS#.exe*
- Add directory name to command search path:
  - **START**, **Control Panel**, **System**
  - Open **Advanced** tab, click on **Environment Variables**
  - Under **System variables**, scroll to variable **PATH**, select it and click on **Edit**
  - Append a semicolon "**;**" followed by the full path to the Executable directory
  - Click **OK**
- Open a DOS-PROMPT window, e.g.,
  - **START**, **All Programs**, **Accessories**, **Command Prompt**
  - Change directory (**cd**) to your working directory with the iTOUGH2 input files, and use the *itough2.bat* file, e.g.:
    `itough2 darcy1i darcy 3`

14

Resources

- iTOUGH2 User's Guide
- iTOUGH2 Sample Problems
- iTOUGH2 Command Reference
- iTOUGH2 Universal Optimization using the PEST Protocol
- iTOUGH2 GSLIB User's Guide
- Parallelization of iTOUGH2 using PVM
- http://esd.lbl.gov/iTOUGH2
- TOUGH Symposia; Short Courses
- Bibliography
- User forum: http://tough.forumbee.com/
- SAFinsterle@lbl.gov

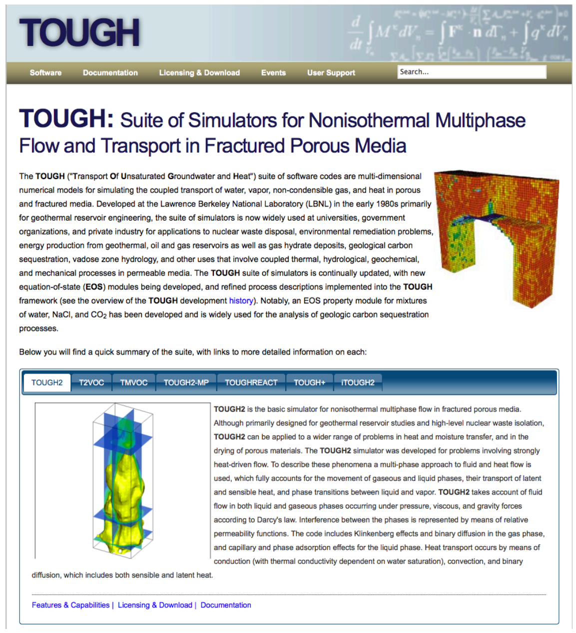
15

Textbooks

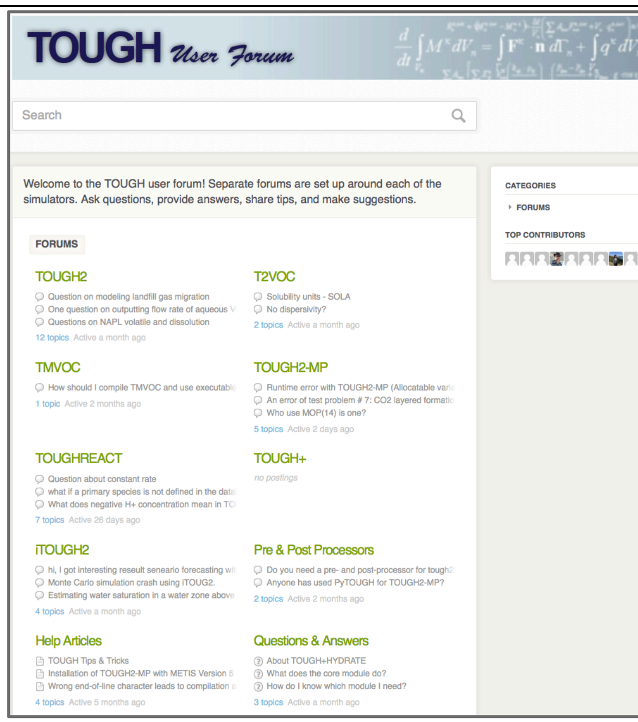
- Aster et al., *Parameter Estimation and Inverse Problems*, 2nd, Ed., Academic Press, 2013.
- Hill and Tiedeman, *Effective Groundwater Model Calibration, With Analysis of Data, Sensitivities, Predictions, and Uncertainty*, Wiley, 2007.
- Saltelli et al., *Global Sensitivity Analysis, The Primer*, Wiley, 2008.
- …

16

http://esd.lbl.gov/TOUGH



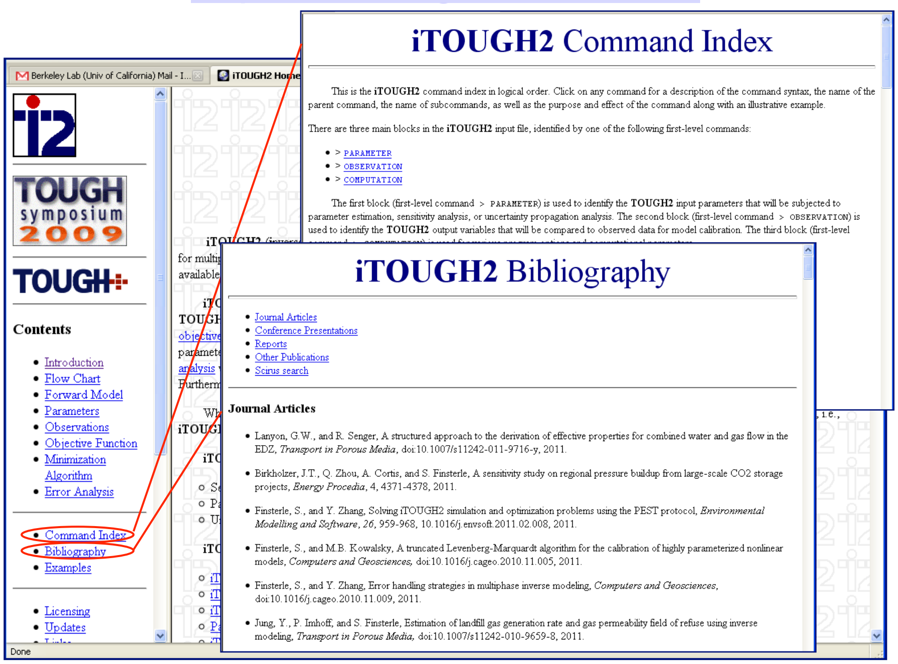
17



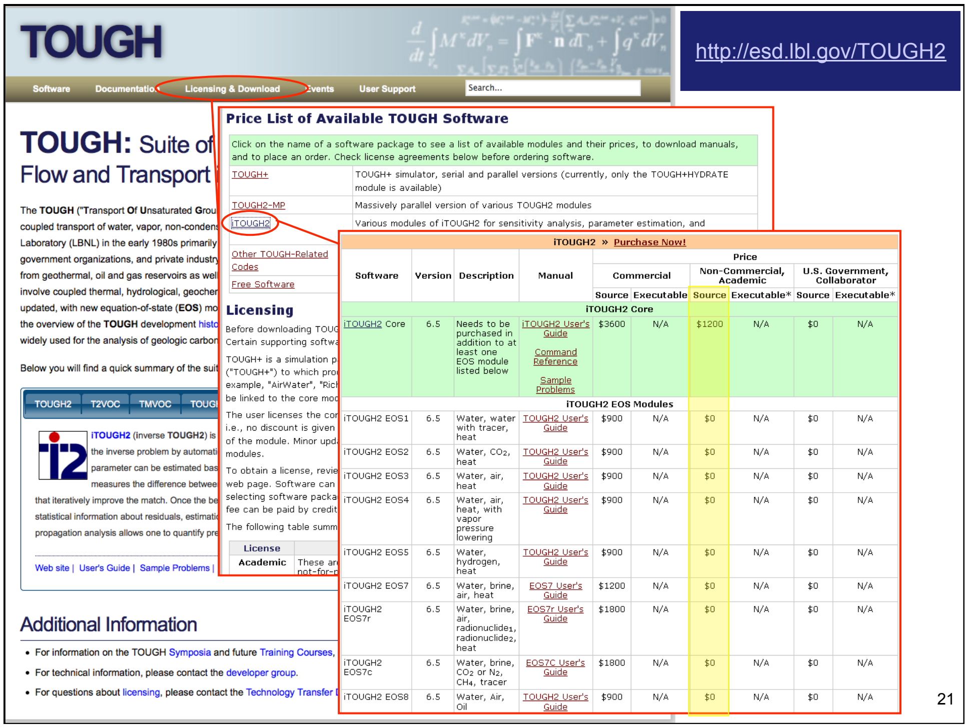
18

19

http://esd.lbl.gov/iTOUGH2



20



http://esd.lbl.gov/TOUGH2

TOUGH

Software | Documentation | Licensing & Download | Events | User Support

Price List of Available TOUGH Software

Click on the name of a software package to see a list of available modules and their prices, to download manuals, and to place an order. Check license agreements below before ordering software.

TOUGH+	TOUGH+ simulator, serial and parallel versions (currently, only the TOUGH+HYDRATE module is available)
TOUGH2-MP	Massively parallel version of various TOUGH2 modules
iTOUGH2	Various modules of iTOUGH2 for sensitivity analysis, parameter estimation, and
Other TOUGH-Related Codes	
Free Software	

Licensing

iTOUGH2 » Purchase Now!

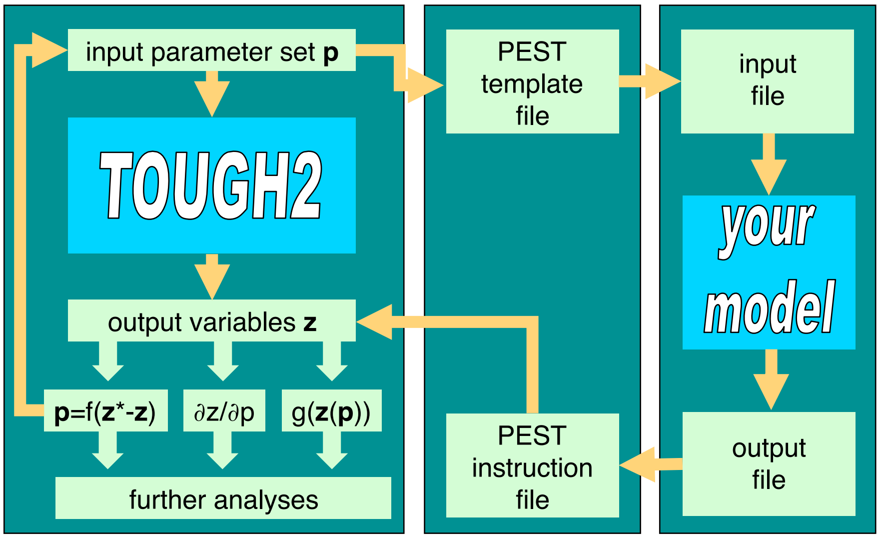
Software	Version	Description	Manual	Price: Commercial Source	Commercial Executable	Non-Commercial, Academic Source	Non-Commercial, Academic Executable*	U.S. Government, Collaborator Source	U.S. Government, Collaborator Executable*
iTOUGH2 Core									
iTOUGH2 Core	6.5	Needs to be purchased in addition to at least one EOS module listed below	iTOUGH2 User's Guide; Command Reference; Sample Problems	$3600	N/A	$1200	N/A	$0	N/A
iTOUGH2 EOS Modules									
iTOUGH2 EOS1	6.5	Water, water with tracer, heat	TOUGH2 User's Guide	$900	N/A	$0	N/A	$0	N/A
iTOUGH2 EOS2	6.5	Water, CO_2, heat	TOUGH2 User's Guide	$900	N/A	$0	N/A	$0	N/A
iTOUGH2 EOS3	6.5	Water, air, heat	TOUGH2 User's Guide	$900	N/A	$0	N/A	$0	N/A
iTOUGH2 EOS4	6.5	Water, air, heat, with vapor pressure lowering	TOUGH2 User's Guide	$900	N/A	$0	N/A	$0	N/A
iTOUGH2 EOS5	6.5	Water, hydrogen, heat	TOUGH2 User's Guide	$900	N/A	$0	N/A	$0	N/A
iTOUGH2 EOS7	6.5	Water, brine, air, heat	EOS7 User's Guide	$1200	N/A	$0	N/A	$0	N/A
iTOUGH2 EOS7r	6.5	Water, brine, air, radionuclide$_1$, radionuclide$_2$, heat	EOS7r User's Guide	$1800	N/A	$0	N/A	$0	N/A
iTOUGH2 EOS7c	6.5	Water, brine, CO_2 or N_2, CH_4, tracer	EOS7C User's Guide	$1800	N/A	$0	N/A	$0	N/A
iTOUGH2 EOS8	6.5	Water, Air, Oil	TOUGH2 User's Guide	$900	N/A	$0	N/A	$0	N/A

Additional Information

- For information on the TOUGH Symposia and future Training Courses,
- For technical information, please contact the developer group.
- For questions about licensing, please contact the Technology Transfer

21

What's good for TOUGH2 may also be good for other codes



23

iTOUGH2-PEST: General Concept

- Forward model ("My_Model") and optimization routines (iTOUGH2) are *separate* codes
- Exchange of information occurs through *ASCII input and output files* using the PEST protocol
- What iTOUGH2-PEST does:
  - Writes forward model *input file*(s) with changed parameter values → *Template File*
  - Calls forward model
  - Extracts select observable variables ("observations") from forward model *output file*(s) → *Instruction File*
- Template and Instruction Files identical to PEST [*Doherty*, 2009; http://www.pesthomepage.org]
- iTOUGH2 input file replaces PEST *Control File*

24

PEST Template File (1 of 4)

- Writes *input* file(s) for `My_Model`, including input from redirected (“<“) *keyboard*
- Identifies *parameters* to be changed by optimization routines
- Same parameter may occur *multiple* times in a template file
- One template file for each input file that contains an adjustable parameter

PEST Template File (2 of 4)

- Copy your input file to *input_file.tpl*
- Add header **`ptf #`**
  - **`ptf`** PEST template file
  - **`#`** parameter delimiter (any special character)
- Replace parameter values by parameter *name* surrounded by parameter delimiters
  - Parameter name: max. 12 character
case insensitive
  - Example: **`Radius = # pipe_radius #`**
- Use same parameter names in **`> PARAMETER`** block of iTOUGH2 input file

PEST Template File Example (3 of 4)

$$y = a_1 + a_2 x + a_3 x^2$$

- Input file of code POLYNOM

```
Polynomial of degree: 2
Coefficient 1: 0.5000000E+00
Coefficient 2: 2.0000000E+00
Coefficient 3: 1.5000000E+00
Evaluate polynomial at  : 5 points
0.25
0.50
1.00
1.50
2.00
```

- Corresponding template file

```
ptf #
Polynomial of degree: 2
Coefficient 1: #coeff0      #
Coefficient 2: #coeff1      #
Coefficient 3: #coeff2      #
Evaluate polynomial at  : 5 points
0.25
0.50
1.00
1.50
2.00
```

27

PEST Template File (4 of 4)

- Relate template file(s) to input files(s) in iTOUGH2 block **> COMPUTATION**
- Example:

```
> COMPUTATION
  >> OPTION
     >>> PEST
         >>>> number of TEMPLATE files: 2
              stdinp.tpl       keyboard
              polynomial.tpl   input.dat
```

28

PEST Instruction File (1 of 7)

- Parses through My_Model *output* file(s) or redirected (“>”) screen output
- Finds and extracts *observable variables*
- Each observable variable may occur *only once* in an Instruction File
- One Instruction File for each output file that contains observable variables

29

PEST Instruction File (2 of 7)

- Header line: **`pif @`**
  - **`pif`** PEST instruction file
  - `@` marker delimiter; marker delimiter must not occur in marker text
- Use *search directives* to find observation
- Instruction lines must start either with a Primary Marker, a Line Advance, or the continuation character (“**`&`**”)
- Instructions pertaining to a single line on a model output file are written on a single line of an instruction file.
- Observation names (max. 20 characters) must be identical to those used in the iTOUGH2 **`> OBSERVATION`** block

30

PEST Instruction File
Search Directives (3 of 7)

Instruction Item	Description	Example Instruction
Primary Marker	Marker at beginning of instruction line. Bracketed by Marker Delimiter	`@OUTPUT@`
Secondary Marker	Marker that does not occupy first instruction item. Searches within current line from left to right. Advances to next line of not found.	`@OUTPUT@ @TIME IS@`
Line Advance	At beginning of instruction line. `L`*n* advances by *n* lines	`l1` `L56`

Observation Name	Unique name identifying observation;
	maximum 20 characters long;
	any ASCII characters except for `[`, `]`, `(`, `)`, or the marker delimiter character.
		`arg1`
		`y2`
		`obs3`
		`pressure_at_X=4`
		`conc-after-5-year`

Dummy Observation	Dummy observation can be used to navigate line by reading non-fixed observations; however, values are not extracted. The observation name for dummy variables must be `dum`.	`l1 !dum! !dum! !sat!`

31

PEST Instruction File
Search Directives (4 of 7)

Instruction Item	Description	Example Instruction
Whitespace	Moves cursor forwards from its current position until it encounters the next blank character, and then moves the cursor forward again until it finds a nonblank character, finally placing the cursor on the blank character preceding this nonblank character.	`@DEPTH =@ w w !p!`
Tab	Places cursor at a user-specified character position on current model output file line.	`@DEPTH =@ t56 !p!`
Fixed Observation	Reads observation between columns *n1* and *n2*. Observation name in brackets; column numbers separated by colon; no spaces	`l1 [pres]13:25`
Semi-Fixed Observation	Reads observation that is contained, starts, or ends somewhere between columns *n1* and *n2*. Observation name in parentheses; column numbers separated by colon; no spaces	`l1 (pres)19:20`

Non-Fixed Observation	Reads observation in free format at current location.
	Observation name between exclamation points.
		`l8 w !pres!`
		`l8 !dum! !dum! !pres!`
		`l5 *=* !sat! *%*`

32

Reading Observations in PEST (5 of 7)

```
(Output File)
----|----1----|----2----|----3----|----4
                   1.87      1.21072
----|----1----|----2----|----3----|----4
```

to extract this number from the output file …

```
pif @ (Instruction File)
Fixed: []
l2 [obs]30:37                [ obs ]

Semi-Fixed: ()
l2 (obs)25:40           (      obs       )
l2 (obs)32:36                 (obs)
l2 (obs)25:32           ( obs  )
l2 (obs)33:40                   ( obs   )

Non-Fixed, free format: !!
l2 w w   !obs!               ! obs !
l2 t27   !obs!             ! obs   !
l2 @.8@  !obs!         !     obs   !
l2 !dum! !obs!         !     obs   !
```

… use one of these options in the instruction file

33

PEST Instruction File Example (6 of 7)

- Output file

```
----|----1----|----2----|----3
Simulation Output File
======================
Iteration 1 Time = 0.2 years
...
Iteration 5 Time = 1.0 years
    Depth            Pressure
    1.00             1.21072
    2.00             1.51313
    3.00             2.07536
    4.00             2.95097
    5.00             4.19023
    6.00             5.87513
    7.00             8.08115
```

- Instruction file

```
pif @
@Iteration@  @1.0 years@
l2 [pres1]20:27
l1 (pres2)11:25
l1 t20   !pres3!
@ 4.00 @ !pres4!
l1 w w   !pres5!
l2 !dum! !pres7!
```

34

PEST Instruction File (7 of 7)

- Relate instruction file(s) to output files(s) in iTOUGH2 block **> COMPUTATION**
- Example:

```
> COMPUTATION
  >> OPTION
     >>> PEST
         >>>> number of INSTRUCTION files: 2
              stdout.ins       screen
              polynomial.ins   output.dat
```

35

Calling Executable from iTOUGH2

- Provide executable, command, script or batch file iTOUGH2 block **> COMPUTATION**
- Use quotes if command consists of multiple words
- Example:

```
  > COMPUTATION
    >> OPTION
       >>> PEST
           >>>> EXECUTABLE: myModel.exe
or
           >>>> EXECUTABLE: Run-ModelA-and-ModelB.bat
or
           >>>> EXECUTABLE: UnixScript.sh
or
           >>>> EXECUTABLE: "a.out < keyboard > screen"
or
           >>>> EXECUTABLE: "octave my_matlab_code.m"
```

36

iTOUGH2 Block
> COMPUTATION, >> OPTION, >>>PEST

```
> COMPUTATION
  >> OPTION
     >>> PEST
         >>>> TEMPLATE: number-of-template-files
              template-file-1.tpl       input-file-1
              ...                       ...
              template-file-ntpl.tpl    input-file-ntpl

         >>>> INSTRUCTION: number-of-instruction-files (NO DELETE)
              instruction-file-1.ins output-file-1
              ...                       ...
              instruction-file-ntpl.ins output-file-nins

         >>>> EXECUTABLE: executable-name  (BEFORE/AFTER)
         >>>> PRECISION : SINGLE/DOUBLE
         >>>> DECPOINT  : NOPOINT/POINT
         <<<<
     <<<
  <<
```

37

iTOUGH2 Block
> PARAMETER, >> PEST

Generic Format

```
> PARAMETER
  >> PEST
     >>> NONE
         >>>> NAME  : parameter-name
         >>>> GUESS : initial-parameter-value   or ...
         >>>> PRIOR : prior-information-value   required
         >>>> other fourth-level commands
         <<<<
     <<<
  <<
```

Example

```
> PARAMETER
  >> PEST
     >>> NONE
         >>>> NAME        : coefficient-A
         >>>> LOGARITHM
         >>>> GUESS       : -3.0
         >>>> RANGE       : -6.0 0.0
         <<<<
     <<<
  <<
```

38

iTOUGH2 Block
> OBSERVATION, >> PEST

Generic Format

```
> OBSERVATION
  (TIMES block not required)
  >> PEST
     >>> UNIVERSAL/MODEL/NONE (: data-set-name)
         >>>> DATA
              observation-name-1     value-1     (weight-1)
              observation-name-2     value-2     (weight-2)
              observation-name-...   value-...   (weight-...)
         >>>> other fourth-level commands
         <<<<
     <<<
  <<
```

39

iTOUGH2 Block
> OBSERVATION, >> PEST

Example

```
> OBSERVATION
  >> PEST
     >>> UNIVERSAL
         >>>> ANNOTATION  :   Total Costs
         >>>> DATA
              capital-cost    0.0
              operating-cost  0.0
         >>>> WEIGHT      :   1078 [KOW/US$]
         <<<<

     >>> UNIVERSAL: pumping rates
         >>>> DATA
              pH-after-0-yr   7.2  1.0  pump
              pH-after-1-yr   5.8  0.5  pump
              pH-after-2-yr   3.6  0.5  pump
         <<<<
      <<<
```

40

Running iTOUGH2-PEST

- If iTOUGH2 is run on a PC *without* TOUGH2 being the forward model, a *dummy TOUGH2 input file* (named, e.g., *dummyT2*) needs to be specified with the keyword **`PEST`** in the first line
- The EOS module "**`pest`**" (or any other EOS module) can be used:

```
itough2 polyi dummyT2 pest
```

- Under Unix/Linux, the dummy TOUGH2 file and dummy EOS name can be replaced by the command line argument **`-pest`**:

```
itough2 -pest polyi
```

41

iTOUGH2-PEST Manual

- Finsterle, S., *iTOUGH2 Universal Optimization Using the PEST Protocol — User's Guide*, Report LBNL-3698E, Lawrence Berkeley National Laboratory, Berkeley, Calif., July 2010.

- http://esd.lbl.gov/research/projects/tough/documentation/manuals.html

42

iTOUGH2 Short Course

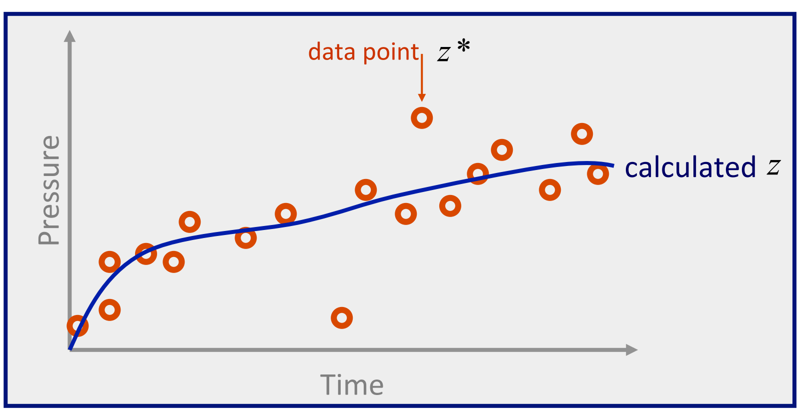
Lawrence Berkeley National Laboratory
Earth Sciences Division
Berkeley, California

Stochastic Model

Types of Errors
Functional vs. Stochastic Models
Observation Covariance Matrix C_{zz}
A Priori Error Variance

Definition: Calibration Point

- A *calibration point* is a point in space and time at which the observed system response z_i^* and the calculated system response z_i will be compared during model calibration.
  - If calibration time does not coincide with measurement time, z_i^* will be linearly interpolated between available measurements



Definition: Residual

- The *residual* is the difference between the observed and calculated system response at calibration point *i.*

$$r_i = z_i * - z(\mathbf{p})_i \qquad i = 1, \ldots, m$$

- The *weighted residual* is the residual multiplied by a weight.
  - An example for weight is the inverse of the assumed measurement error (stochastic model).

$$y_i = \frac{z_i * - z(\mathbf{p})_i}{\sigma_i} \qquad i = 1, \ldots, m$$

3

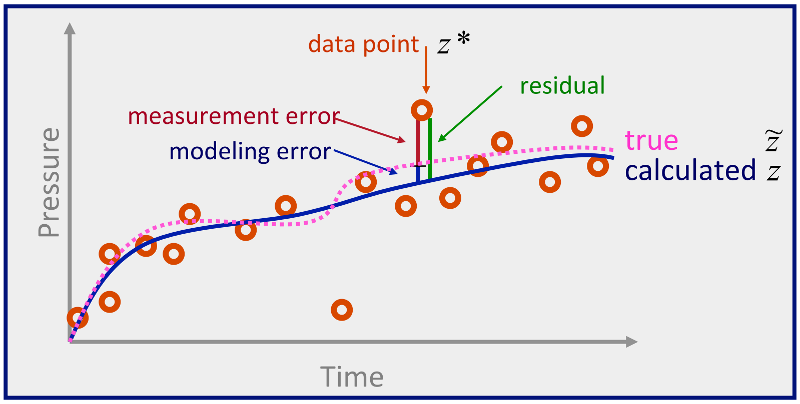
Definition: Jacobian Matrix

- The Jacobian **J** is an $m \times n$ matrix holding the *sensitivity coefficients*
- The local sensitivity coefficients are the *partial derivatives* of the calculated system response z_i at all calibration points z_i, $i = 1, \ldots, m$, with respect to each of the parameters p_j, $j = 1, \ldots, n$

$$J_{ij} = \frac{\partial z(\mathbf{p})_i}{\partial p_j}$$

4

Errors and Residual



- Measurement error $e_d = z^* - \tilde{z}$
- Modeling error $e_m = z - \tilde{z}$
- Residual $r = e_d - e_m = (z^* - \tilde{z}) - (z - \tilde{z}) = z^* - z$

5

Systematic and Random Components

data point	=	true system state	+	measurement error
≀		systematic identified conceptual functional		random unidentified distributional stochastic
calibration point	=	calculated system state	+	residual

6

Functional vs. Stochastic Models

- Functional Model
  - Attempts to capture the systematic, identifiable aspects
    - True system state
    - Systematic error in the data
  - Is represented by the governing equations of the forward model
- Stochastic Model
  - Describes random, unidentifiable aspects
  - Includes:
    - A distributional assumption about the final residuals
    - Estimate of expected size of residuals (*not* measurement errors!)
  - Is represented by the observation covariance matrix $\mathbf{C}_{zz}$
  - Describes measurement error only if Functional Model is perfect

7

Stochastic Model Development

Before inversion (*a priori*):

- Estimate *expected distribution of residuals* (type of distribution and standard deviation).
  - Consider which portion of the observed signal shall be explained by the functional model.
  - Consider *measurement errors* and *modeling errors*.
  - Consider only *random* components.
  - Talk to experimentalist/data collector/data analyst!
- Set up observation covariance matrix $\mathbf{C}_{zz}$ (or $\sigma_0^2\,\mathbf{V}_{zz}$)

8

Observation Covariance Matrix $\mathbf{C}_{zz}$

- $m \times m$ diagonal matrix

$$\mathbf{C}_{zz} = \begin{bmatrix} \sigma_1^2 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \sigma_i^2 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \sigma_n^2 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \sigma_j^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \cdots & \sigma_m^2 \end{bmatrix} = \sigma_0^2 \cdot \mathbf{V}_{zz}$$

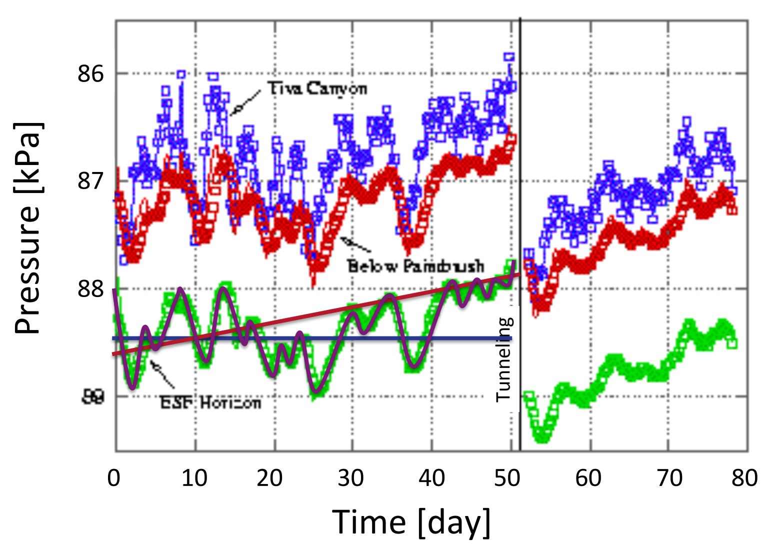
σ_0^2 : a priori error variance

In iTOUGH2
σ_0^2 = 1.0
V_{zz} = C_{zz}

- Summarizes the Stochastic Model
- Scales data of different quality, type, and units of measurement
- Weights fitting error
- Reflects your *prior* assumption about the average size of the residual *after* calibration

9

Exercise: Determine $\mathbf{C}_{zz}$



10

iTOUGH2 Input

One alternative to specify *a priori* variance to all data in a given data set:

```
> OBSERVATION
  >> PRESSURE
     >>> ELEMENT                   : A1125
         >>>> ANNOTATION           : Pressure 1/2
         >>>> COLUMNS              : 1 2
         >>>> Read DATA from FILE  : nonoise.dat (in MINUTES)
         >>>> standard DEVIATION   : 200.0 Pa     (expected residuals)
         <<<<
     <<<
  <<
```

11

Prior Stochastic Model for Parameters

- Parameter uncertainty distributions needed for
  - parameter estimation with prior information
  - uncertainty propagation analysis
  - sensitivity analysis (scaling of composite SA measures)

```
> PARAMETER
  >> ABSOLUTE permeability
     >>> MATERIAL                  : SAND_ BOUND
         >>>> ANNOTATION           : log(abs. perm.)
         >>>> LOGARITHM
         >>>> Log-NORMAL distribution of UQ
         >>>> PRIOR information    : -12.0   (mean for UQ)
         >>>> standard DEVIATION :    0.5   (weighs difference
                         between estimate and prior value)
         >>>> BOUNDS               : -13.5 -10.5
         <<<<
     <<<
  <<
```

12

Example: darcy2i

```
+!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
ESTIMATED PARAMETER  V/L/F   ROCKS    PAR  INITIAL GUESS    BEST ESTIMATE

log(abs. perm.)      LOG10   SAND +1   1   -0.12000E+02      -0.1169E+02
Porosity             VALUE   SAND      1    0.25000E+00        0.373E+00
Initial gas sat      VALUE   DEFAU     2    0.10250E+02      0.10291E+02
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!



!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
       STANDARD DEVIATIONS            SENSITIVITY
 A PRIORI     MARGINAL      C/M     OUTPUT    OBJ. FUNC.
      N/A    0.169E-01     0.505     333.5         0.302
      N/A    0.170E-01     0.455      33.9         0.279
      N/A    0.408E-02     0.820      85.2         0.905
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

`>>>> DEVIATION` in block `> PARAMETER` not specified in darcy2i:

→ difference between estimated parameter and its prior value is *not* weighted

→ "prior information" is not included

13

a posteriori

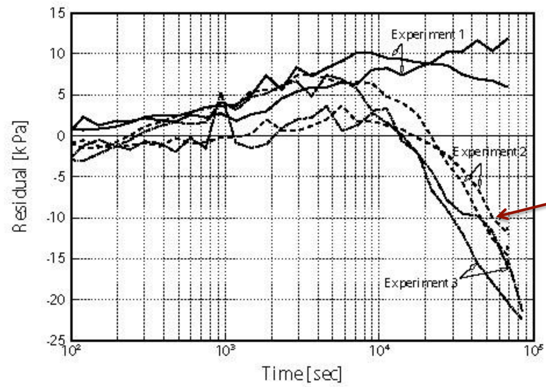
After inversion (*a posteriori*):

- Determine if the final residuals are consistent with the (*a priori* determined) stochastic model
- Test randomness of residuals (look for systematic structure/bias in residuals) → see *Residual Analysis*
- Perform Fisher Model Test (tests *a priori* error variance σ_0^2 against *a posteriori* error variance $\boldsymbol{s}_0^2$)

14

Stochastic Model Development

Leakage in experimental apparatus leads to systematic errors



Estimated error variance s_0^2=48.31

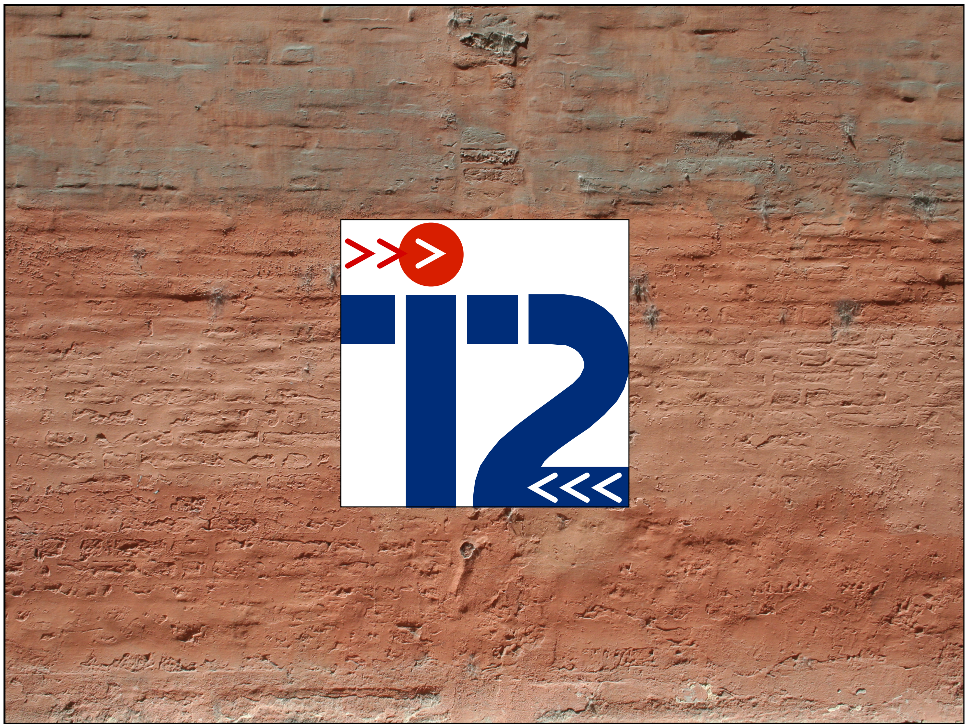
$$s_0^2 = \frac{\mathbf{r}^T \mathbf{C}_{zz}^{-1} \mathbf{r}}{m-n}$$

Systematic error

Fisher test: $s_0^2/\sigma_0^2 > 1$. Error in functional model

Stochastic Model: Questions

1. List types of errors?
2. Provide examples of systematic errors?
3. Provide examples of random errors?
4. How are random errors described?
5. What is the purpose of the Stochastic Model?
6. Why do we often base the Stochastic Model on measurement errors?
7. What is the value of σ_0^2?
8. What is the distribution of a sum of many random errors?



Alternative ways to specify σ

All data points in data set receive same σ

```
>>>> DEVIATION: 200    Pa      σ
>>>> VARIANCE : 4E4    Pa^2    σ²
>>>> WEIGHT   : 0.005 1/Pa     1/σ
>>>> AUTOMATIC                 σ = 10% of mean of all
                               measured data in data set
```

Each data point receives its own σ

```
>>>> RELATIVE : 2 %            σ = 2% of measured value

>>>> COLUMNS  : 1 2 3
>>>> DATA          time       value       std. dev.
                 3600.0     10310.5        200.0
                86400.0     55195.0        400.0
```

18

σ for Parameters

Weight of prior information and scaling of sensitivity coefficients

`>>>> DEVIATION: 1.0` $\log(m^2)$ prior information
`>>>> VARIATION: 1.0` $\log(m^2)$ variation for SA

Parameter uncertainty distributions for Monte Carlo analyses

```
>>>> UNIFORM
>>>> NORMAL
```

Combine with parameter transformation commands
`>>>> LOGARITHM`, `>>>> FACTOR`, `>>>> LOG(F)`
as well as commands `>>>> GUESS/PRIOR` and `>>>> RANGE`
to fully define uncertainty distributions

19

Related Commands

Specify individual diagonal elements of covariance matrix

```
>> COVARIANCE
   32   500.0
```

σ = 500 Pa for calibration point No. 32

Accounting for autocorrelated and heteroscedastic residuals

`>>>> AUTOREGRESSION`	Set AR1 autocorrelation coefficient ρ
`>> AUTOREGRESSION`	Estimate AR1 autocorrelation coefficient ρ
`>>>> Box-Cox`	Set Box-Cox transformation parameter λ
`>> Box-Cox`	Estimate Box-Cox transformation parameter λ

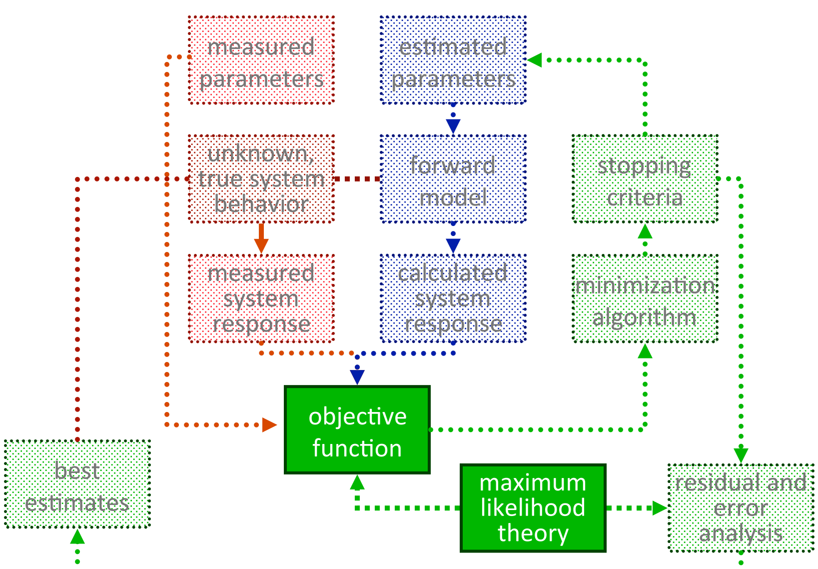
20

iTOUGH2 Short Course

Lawrence Berkeley National Laboratory
Earth Sciences Division
Berkeley, California

Objective Function

Measure of Misfit
Maximum Likelihood
Least Squares
Robust Estimators



2

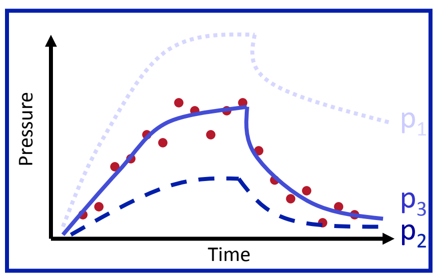
Objective Function

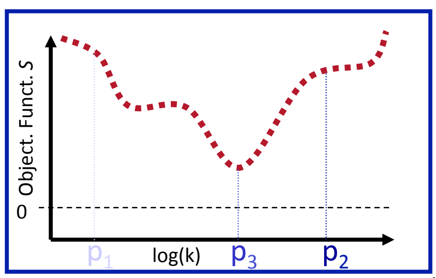
- Aggregate *measure of misfit* between measured data and model prediction
- Scalar, S
- Function of unknown parameters $\mathbf{p}$: $S=S(\mathbf{p},\mathbf{z}(\mathbf{p}))$
- To be minimized (*minimizing S = improving fit*)
- Can be based on maximum likelihood considerations
- Examples of objective functions:
  - Least absolute value (L_1)
  - *Least squares* (L_2)
  - MinMax (L_∞)
  - Robust estimators (Cauchy, Huber, Andrews)
  - Others (Nash-Sutcliffe, Kling-Gupta)

3

Objective Function: Aggregate Measure of Misfit

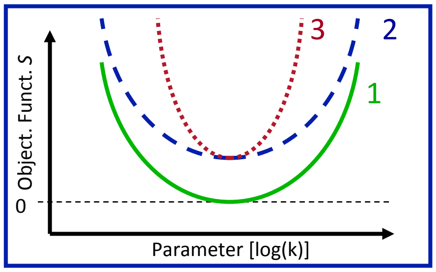
- Pressure data from well test
- Predicted pressures depend on parameter log(k)
- Objective function S is aggregate measure of misfit
- S is function of log(k)
- log(k) that minimizes S yields best fit
- p_3 is considered best estimate of log(k)
- p_3 is parameter that most likely "produced" the observed pressure data.

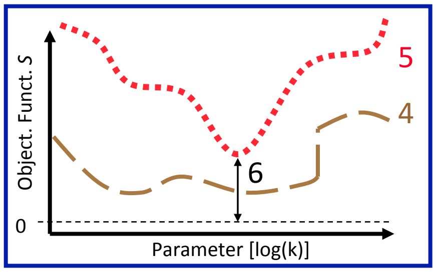




4

Objective Function for *n* = 1





1

2

3

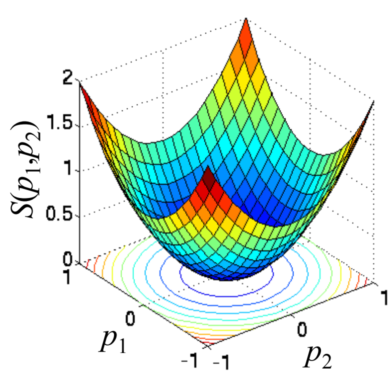
4

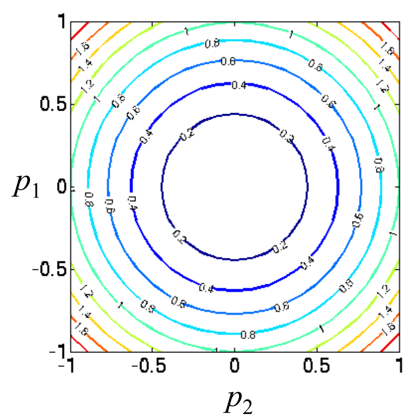
5

6

5

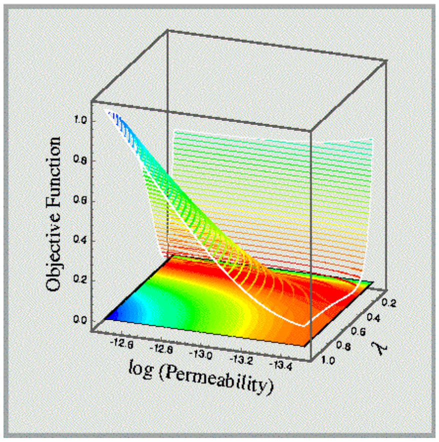
Objective Function for 2 Parameters

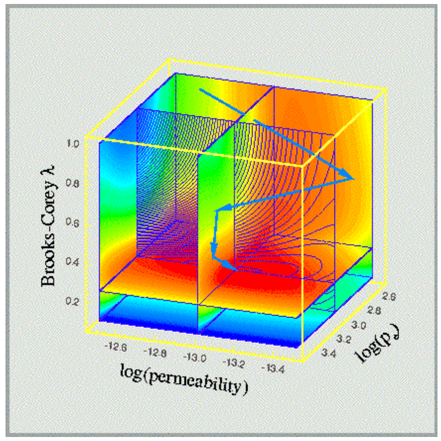




6

Objective Function for *n*=2 and *n*=3





7

Properties of Objective Function

- A *nonlinear* function in *n*-dimensional parameter space

Ill-posed

- *Non-convex,* with many local minima, saddle points, long and narrow valleys, etc.
- Discontinuous and numerically unstable.
- Flat near the minimum.

Well-posed

- *Convex*, but may exhibit local minima and saddle points.
- *Continuous* and *differentiable.*
- Close to *quadratic* near minimum.

8

Weighted Least Squares

- Defined as the sum of squared residuals, multiplied by an appropriate weight

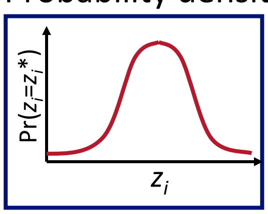
$$\textbf{minimize} \quad S = \sum_{i=1}^{m} \frac{(z_i * - z(\mathbf{p})_i)^2}{\sigma^2}$$

- Most common
- Increasing the weight increases the contribution of that observation to the objective function.
- Least squares estimation (by definition) minimizes error variance

9

Maximum Likelihood: General

- Probability density function (pdf)



– Probability of observing **z*** if **p** is true.
– Joint pdf is the product of individual observations' pdf assuming the observations are independent.

$$\Phi(\mathbf{z};\mathbf{p}) = \Pr(\mathbf{z} = \mathbf{z}^* \mid \mathbf{p}) = \prod \Phi(z_i;\mathbf{p})$$

- Likelihood function
  - Describes the likelihood of **p** given **z***.
  - Same concept: $\Phi(\mathbf{z};\mathbf{p}) \Leftrightarrow L(\mathbf{p};\mathbf{z}^*)$
  - What do we want to do if we want to find **p** that best fit **z***?
- Maximize the likelihood of the solution

$$\text{maximize} \;\; L(\mathbf{p};\mathbf{z}^*) \Leftrightarrow \text{minimize} \;\; S = -2\ln\left[L(\mathbf{p};\mathbf{z}^*) \right]$$

10

Maximum Likelihood: Normal Distribution = Least Squares

- Gaussian probability density function

$$\Phi(z_i;\mathbf{p}) = (2\pi\sigma_i)^{-1/2}\exp\left[-\frac{1}{2}\frac{(z_i*-\hat{z}_i)^2}{\sigma_i^2}\right]$$

$$\Phi(\mathbf{z};\mathbf{p}) = \prod_{i=1}^{m}\Phi(z_i;\mathbf{p}) = (2\pi)^{-m/2}\left|\mathbf{C}_{zz}\right|^{-1/2}\exp\left[-\frac{1}{2}(\mathbf{z}*-\hat{\mathbf{z}})^T\mathbf{C}_{zz}^{-1}(\mathbf{z}*-\hat{\mathbf{z}})\right]$$

- Likelihood function

$$L(\mathbf{p};\mathbf{z}*) = (2\pi)^{-m/2}\left|\mathbf{C}_{zz}\right|^{-1/2}\exp\left[-\frac{1}{2}(\mathbf{z}*-\hat{\mathbf{z}})^T\mathbf{C}_{zz}^{-1}(\mathbf{z}*-\hat{\mathbf{z}})\right]$$

- Maximum likelihood $\quad$ **minimize** $S = (\mathbf{z}*-\mathbf{z})^T\mathbf{C}_{zz}^{-1}(\mathbf{z}*-\mathbf{z})$

Least square minimization !

11

Weighted Least Squares

- IF residuals are normally distributed
  - A reasonable assumption based on central limit theorem, assuming the observations are independent.
- IF weights are the inverse of the standard deviations
- THEN minimizing the least-squares objective function yields *maximum-likelihood estimates*.

$$\textbf{minimize} \quad S = \sum_{i=1}^{m}\frac{(z_i*-z_i)^2}{\sigma_i^2} = (\mathbf{z}*-\mathbf{z})^T\mathbf{C}_{zz}^{-1}(\mathbf{z}*-\mathbf{z})$$

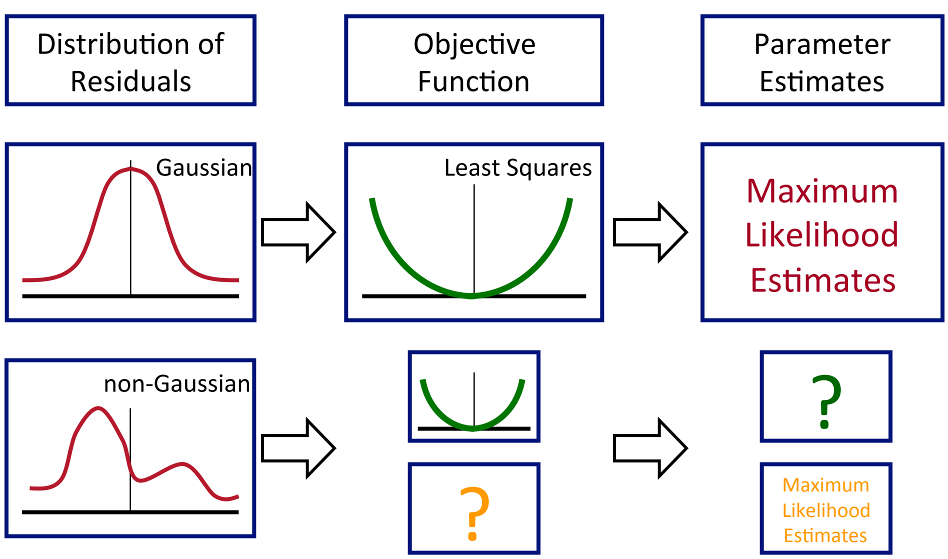
12

Weighted Least Squares

- This interpretation is only valid if:
  – Residual errors are random and independent
  – Residual errors are normally distributed
  – No systematic error
- Violations of the assumptions:
  – Presence of few large outliers
    - Small number of deviate points can distort fit
  – Presence of many small outliers
    - Error distribution is heavy tailed
  – Systematic errors
    - Asymmetric distribution
    - Deterministic instead of random
    - Correlated residuals

13

Non-Gaussian Residuals



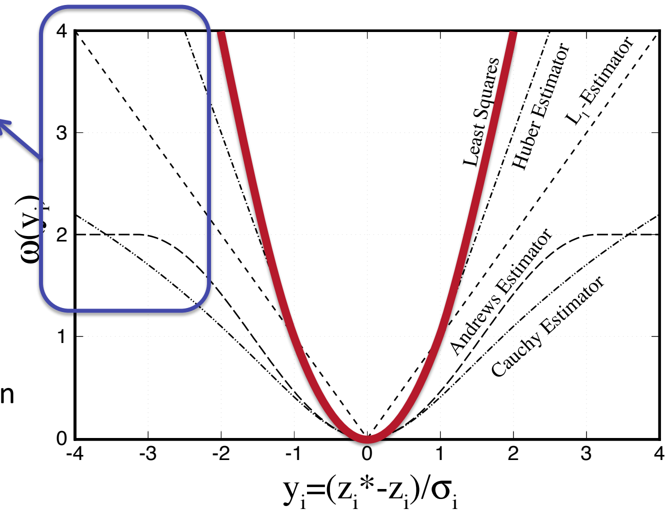
14

Robust Estimators

Reduced impact of large residual.

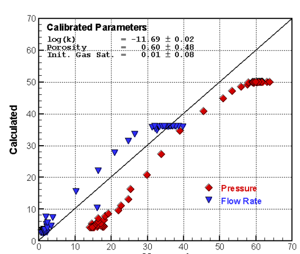
$$S = \sum_i \omega(y_i)$$

loss function



Large residuals can be due to outliers in the data or systematic modeling/data errors

15



heteroscedastic

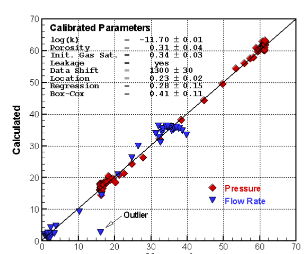
$$\tilde{z}(z;\lambda) = \begin{cases} \dfrac{(z^{\lambda}-1)}{\lambda \cdot g^{\lambda-1}} & \text{if } \lambda \neq 0 \\ g \cdot \ln(z) & \text{if } \lambda = 0 \end{cases}$$

autocorrelated

$$\tilde{r}_i = \begin{cases} r_1\sqrt{(1-\rho^2)} & \text{if } i=1 \\ r_i - \rho \cdot r_{i-1} & \text{if } i=2,\ldots,m \end{cases}$$

skewed and heavy-tailed

$$p(r_i \mid \xi, \beta) = \frac{2\sigma_{\xi}}{\xi+\xi^{-1}}\omega_{\beta}\exp\left\{-c_{\beta}\left|r_{\xi,i}\right|^{2/(1+\beta)}\right\}$$



concurrently estimate *statistical* parameters
β, λ, ρ, ξ

Schoups and Vrugt, *WRR*, 2010; Finsterle and Zhang, *C&G*, 2011

"Theoria Cominationis Observationum Erroribus Minimis Obnoxiae" Carl Friedrich Gauss (~1820)

The integral ∫xϕx.dx, i.e., the mean value of *x*, indicates the presence or absence of constant error, as well as its magnitude. Similarly, the integral ∫xxϕx.dx taken from x=-∞ to x=+ ∞ (the mean square of *x*) **seems most appropriate** to generally define and quantify the uncertainty of the observations. Thus, given two systems of observations which differ in their likelihoods, we will say that the one for which the integral ∫xxϕx.dx is smaller is the more precise.

Now if someone should object that this convention has been chosen arbitrarily with no compelling necessity, I will gladly agree. In fact, the problem has an intrinsic vagueness about it that can only be resolved by a more or less **arbitrary principle**. It is not out of place to compare the estimation of quantity by means of an observation subject to larger or smaller errors with a game of chance. Since any error to be feared in an observation is connected with a loss, the game in one in which nobody wins and everybody looses. We estimate the outcome of such a game from the probable loss: namely, from the sum of the product of the individual losses with their respective probabilities.

It is by no means self-evident how much loss should be assigned to a given observation error. On the contrary, the matter depends in some part on our own judgment. Clearly we cannot set the loss equal to the error itself; for if positive errors were taken as losses, negative errors would have to represent gains. The size of the loss is better represented by a function that is naturally positive. Since the number of such functions is infinite, it would seem that we should **choose the simplest function** having this property. **That function is arguably the square**, since the principle proposed above results from its adoption.

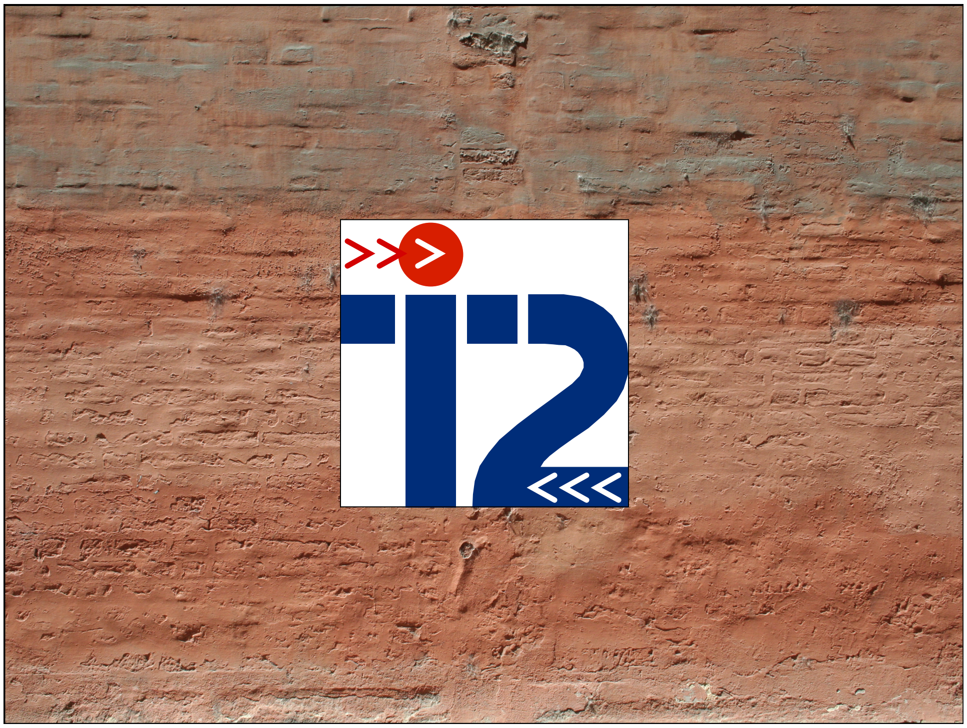
Laplace has also considered the problem in a similar manner, but he adopted the **absolute value of the error** as his measure of loss. Now if I am not mistaken, **this convention is no less arbitrary than mine**. Should an error of double size be considered as tolerable as a single error twice repeated or worse? Is it better to assign only twice as much influence to a double error or more? T**he answers are not self-evident, and the problem cannot be resolved by mathematical proofs, but only by an arbitrary decision**. Moreover, it cannot be denied that Laplace's convention violates continuity and hence resists analytic treatment, while the results that my convention leads to are distinguished by their **wonderful simplicity and generality**.

("Theory of the Combination of Observations Least Subject to Errors", translated from Latin by G.W. Stewart, SIAM, 1995; emphases added) 17

Objective Function: Questions

1. Purpose of objective function?
2. Properties of objective function?
3. Well-posed inverse problem?
4. Ill-posed inverse problem?
5. Reasons for choosing least-squares?
6. Potential problems with least-squares?
7. Sketch contours of objective function (*n*=2) for nonlinear model, well-posed inverse problem, noisy data, correlated parameters

18



iTOUGH2 Commands
Objective Function

```
> COMPUTATION

  >> OPTION
```

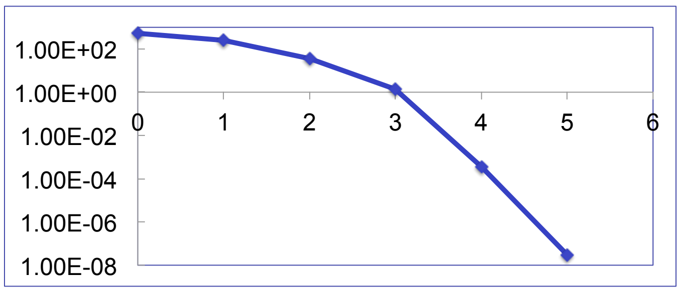
Objective function options:

```
    >>> LEAST-SQUARES
    >>> ANDREWS: 1.5
    >>> CAUCHY
    >>> L1-ESTIMATOR
    >>> QUADRATIC-LINEAR: 2.0
    >>> NASH-SUTCLIFFE
    >>> KLING-GUPTA
    >>> SEP
```

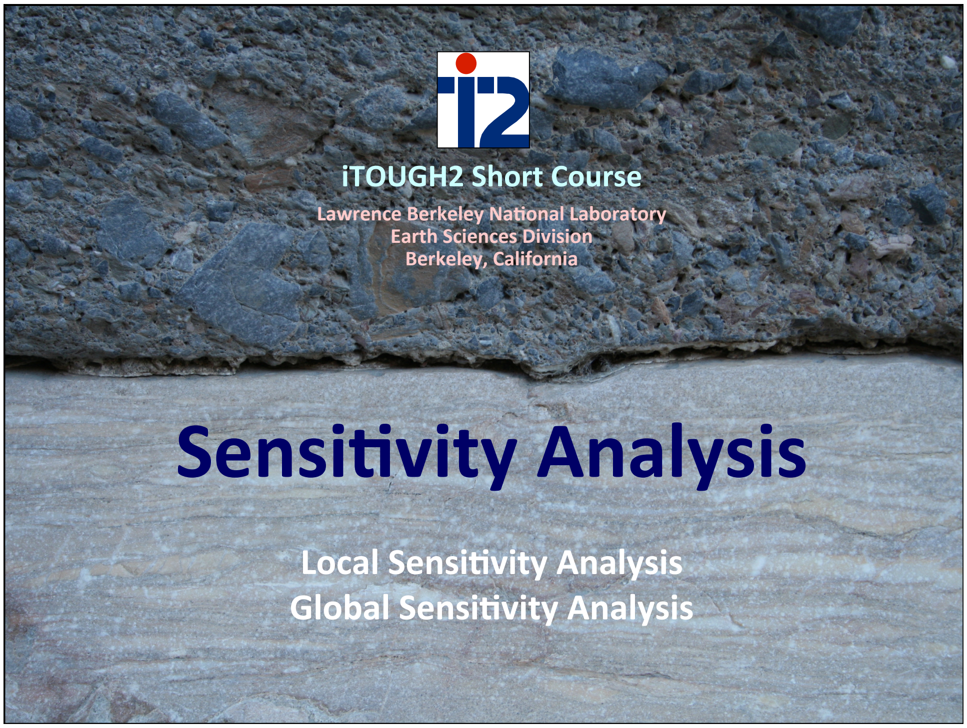
20

Example: darcy2i

```
------------------------------------
ITER TOUGH2 OBJ FUNC. MAX. RESID.
------------------------------------
>I   0     1 0.54881E+03 0.46643E+02
>I   1     5 0.25961E+03 0.32326E+02
>I   2     9 0.36443E+02 0.50984E+01
>I   3    13 0.14103E+01 0.10593E+00
>I   4    17 0.36144E-03 0.30614E-04
>I   5    21 0.30869E-07 0.27656E-08
```



21



Definitions

- *Sensitivity Analysis*:

  "*The study of how uncertainty in the output of a model can be apportioned to different sources of uncertainty in the model input*"

  (Saltelli et al., 2004)

- *Uncertainty Quantification / Analysis*:

  Focuses on *quantifying* uncertainty in model output given uncertainties in model input

- Observations are *sensitive / insensitive*

- Parameters are *influential / non-influential*

2

Purpose of Sensitivity Analysis

- Sensitivity Analysis provides insight into:
  - *System behavior*, *features* and *processes:*
    - Helps uncover errors in law-driven models
    - Helps build parsimonious data-driven models
    - Helps identify key processes in diagnostic models
    - Helps identify key factors affecting prognostic models
    - Helps defend robustness of model
    - Helps establish research priorities
  - Relative *importance of parameters* →
    - Which (uncertain) parameters have greatest effect on model predictions and prediction uncertainties?
    - Which properties need to be determined with high accuracy?

3

Purpose of Sensitivity Analysis

- Information content of data ("*worth of data*")
  - Which data contain information about the parameters to be estimated by inverse modeling?
  - Which parameters may be estimated using inverse modeling?
- Sensitivity always refers to a *specific objective*.
- Sensitivity measures are to be consider *qualitative* or (at best) semi-quantitative.

4

Types of Sensitivity Analyses

- Sensitivity of *z* with respect to *p*
  - Observable variables w.r.t. parameters to be estimated
  - Objective function w.r.t. parameters to be estimated
  - Performance measures w.r.t. design variables
  - Model output w.r.t. uncertain input parameters
- Consider potential variation of parameter *p*
- Consider expected/acceptable variability of observable *z*
- Make sensitivity coefficients dimensionless for comparison

5

Types of Sensitivity Analyses

- *Local* Sensitivity Analysis
  - Sensitivity *at a given point* in the parameter space
  - Byproduct of derivative-based minimization algorithms
  - Analytical or numerical evaluation

- *Global* Sensitivity Analysis
  - Composite sensitivity measure *over feasible parameter domain*
  - Global SA only needed if model is (highly) nonlinear
  - Sampling-based evaluation

6

Sensitivity Matrix

- The $m \times n$ Jacobian **J** matrix holds the local sensitivity coefficients
- The local sensitivity coefficients are the partial derivatives of the calculated system response z_i, $i = 1, \ldots, m$, with respect the parameters p_j, $j = 1,\ldots,n$.

$$J_{ij} = \frac{\partial z(\mathbf{p})_i}{\partial p_j}$$

- The sensitivity coefficients can be scaled by the expected variations of the parameters and observations → dimensionless

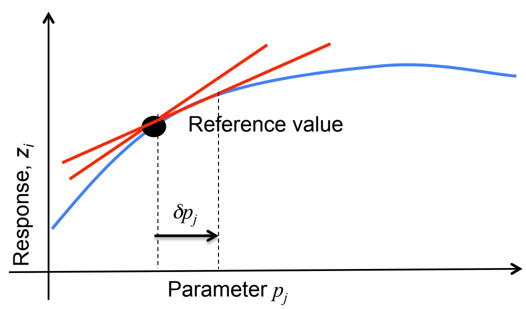
$$\tilde{J}_{ij} = \frac{\partial z(\mathbf{p})_i}{\partial p_j} \cdot \frac{\sigma_{p_j}}{\sigma_{z_i}}$$

8

Sensitivity Matrix

- Partial derivatives are evaluated numerically
- Default perturbation in iTOUGH2: $\delta p = 0.01p$
- Local sensitivities depend on reference parameter set

$$J_{ij} = \left.\frac{\partial z_i}{\partial p_j}\right|_{p_j^*} \approx \frac{z_i(p_1,\ldots,p_j+\delta p_j,\ldots,p_n) - z_i(\mathbf{p})}{\delta p_j}$$



9

Composite Sensitivity Measures (1 of 2)

- Relative sensitivity of each data point to each parameter:

$$\tilde{J}_{ij} = J_{ij}\frac{\sigma_{p_j}}{\sigma_{z_i}} = \frac{\partial z_i}{\partial p_j}\cdot\frac{\sigma_{p_j}}{\sigma_{z_i}}$$

- Information content of individual data points:

$$a_i = \sum_{j=1}^{n}\left|\tilde{J}_{ij}\right|$$

- Overall parameter sensitivity:

$$d_j = \sum_{i=1}^{m}\left|\tilde{J}_{ij}\right|$$

	Parameter j				Σ
Observation	$\tilde J_{11}$	$\tilde J_{1j}$	…	$\tilde J_{1n}$	a_1
	$\tilde J_{i1}$	$\tilde J_{ij}$	…	$\tilde J_{in}$	a_i
	⋮	⋮		⋮	⋮
	⋮	⋮		⋮	⋮
i	$\tilde J_{m1}$	$\tilde J_{mj}$	…	$\tilde J_{mn}$	a_m
Σ	d_1	d_j	…	d_n	

10

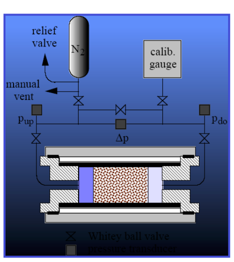
Composite Sensitivity Measures (2 of 2)

- Information content of individual data set to estimation of a parameter:
  $$b_{kj} = \sum_{i=1}^{m} \left| \tilde{J}_{ij} \right| \Big|_{i \in k}$$
- Information content of individual data set to all parameters:
  $$c_k = \sum_{j=1}^{n} \sum_{i=1}^{m} \left| \tilde{J}_{ij} \right| \Big|_{i \in k} = \sum_{i=1}^{m} a_i \Big|_{i \in k} = \sum_{j=1}^{n} b_{kj}$$
- Parameter influence on objective function:
  $$\delta_j = |\Delta S| = |S(p_i + \Delta p_i) - S(p_i)|$$
- Measures do not account for correlations among data and/or parameters

	Parameter j				Σ
Data set	b_{11}	b_{1j}	$\cdots$	b_{1n}	c_1
	b_{k1}	$\boxed{b_{kj}}$	$\cdots$	b_{kn}	c_k
	$\vdots$	$\vdots$		$\vdots$	$\vdots$
k	b_{K1}	b_{Kj}	$\cdots$	b_{Kn}	c_K
Σ	d_1	d_j	$\cdots$	d_n	

Example Local Sensitivity Analysis

The image cannot be displayed. Your computer may not have enough memory to open the image, or the image may have been corrupted. Restart your computer, and then open the file again. If the red x still appears, you may have to delete the image and then insert it again.



Finsterle, S., and P. Persoff, *Water Resour. Res.*, *33* (8), 1803–1811, 1997.

Example: darcy2i

```
********************************
SENSITIVITY ANALYSIS
********************************
```

Look for SA in darcy2i.out

a_i, information content of individual data

[… descriptions of sensitivity analysis …]

```
      TIME # OBSERVATION  log(abs. perm.)     Porosity Initial gas sat      TOTAL
5.00000E-01 4  Pressure 1/2   -1.65294E+00 1.84821E-01 -2.79597E+00  4.63373E+00
1.00000E+00 6  Pressure 1/2   -4.89020E+00 5.65301E-01 -4.08169E+00  9.53720E+00
1.50000E+00 8  Pressure 1/2   -5.47909E+00 6.35593E-01 -4.36425E+00  1.04789E+01
```

[… repeated for different TIME and OBSERVATION …]

information content of individual data set

```
_____________________________________________________
log(abs. perm.)      Porosity Initial gas sat      TOTAL
---------------------------------------------------------------------------
Sum of sensitivity coefficients 6.67235E+02  6.75808E+02  1.69667E+03
Potential parameter variation   5.00000E-01  5.00000E-02  5.00000E-02
----------------------------------------------------------------------------
Total from data  Pressure 1/2         239.1         27.8          64.9    331.8
Total from data  Flow inlet            94.5          6.0          19.9    120.4
----------------------------------------------------------------------------
Total parameter sensitivity           333.6         33.8          84.8
============================================================================
```

d_i, overall parameter sensitivity

c_k, overall information content of data set

13

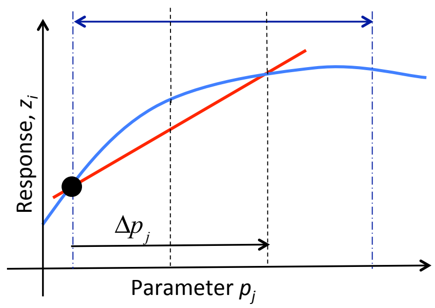
Global Sensitivity Analysis

Morris OAT

Variance-Based Sensitivity Analysis

Morris One-At-a-Time (MOAT)

- Comprised of multiple local sensitivity analyses
- Partition parameter space in *k* points (*k*-1 intervals)
- Define $\Delta p_j = \dfrac{k}{2(k-1)}$
- Change parameters by Δp_j, *one at a time*, and compute finite difference sensitivity coefficients



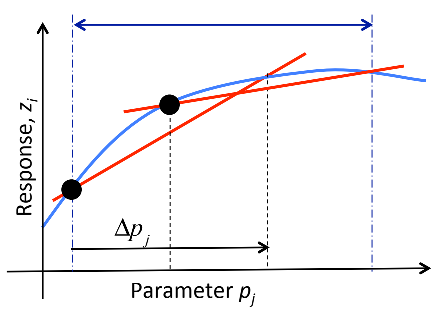
➡ *elementary effects*

$$S_{ij} = \frac{z_i(p_1, \ldots p_{j-1}, p_j + \Delta p_j, p_{j-1}, \ldots, p_n) - z_i(\mathbf{p})}{\Delta p_j}$$

15

MOAT

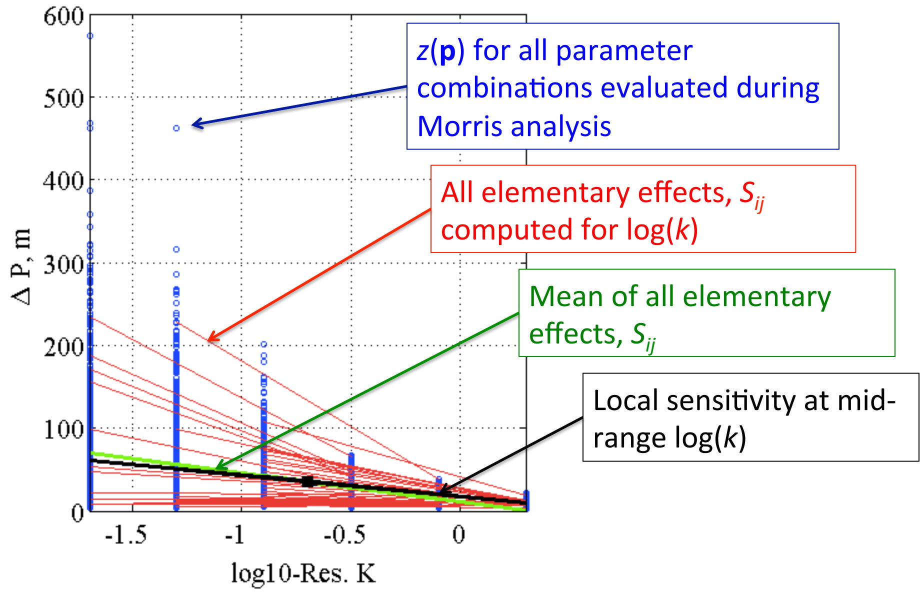
- Evaluate S_{ij} for multiple *paths* (n_p) in the parameter space, each starting at a randomly selected point



- Calculate *mean elementary effect*: $\mu_S = \mathbf{E}[S_{ij}(\mathbf{p})]$
- Calculate *mean absolute elementary effect*: $\mu_S{}^* = \mathbf{E}[|S_{ij}(\mathbf{p})|]$
- Calculate *std. dev. of elementary effect*: $\sigma_S^2 = \mathbf{V}[S_{ij}(\mathbf{p})]$
- Number of simulations needed: $N_M = (n+1) \bullet n_p$

16

MOAT visualized



17

MOAT

- Morris OAT method identifies parameters that are:
  - Important / negligible
  - Linear / nonlinear
  - Independent / correlated
- Possible scenarios:

  (1) Linear model z(**p**) → S_{ij}(**p**) constant → $\sigma_S^2 = 0$

  (2) p_j has negligible effect on z_i → $\mu_S^* \approx 0$

  (3) Nonlinear model z(**p**) → S_{ij}(**p**) variable → $\sigma_S^2 > 0$

  (4) Interactions among p's → S_{ij}(**p**) variable → $\sigma_S^2 > 0$
- Therefore:

  (1) If $\mu_{S_j}^* >> 0$, then parameter p_j is important

  (2) If , then parameter p_j is insignificant

  (3) If $\sigma_S^2 > 0$ then p_j has nonlinear/interaction effect

18

MOAT

- Demonstration for linear/nonlinear model

$$z(\mathbf{p}) = p_1^2 + 2p_2 + 3p_3 + 4p_4 + 5p_5$$

	p_1	p_2	p_3	p_4	p_5
μ_{S_j}					
$\sigma_{S_j}^2$					

- Parameters p_2 - p_5 are linear:
  - The mean elementary effects correspond to the local sensitivity values
  - Variances are zero
- Parameter p_1 has nonlinear effect:
  - For $0 < p_j < 1$, the mean elementary effect is < 1
  - Variance is non-zero

19

MOAT

- Demonstration for model with interaction

$$z(\mathbf{p}) = p_1 \cdot p_2 + 3p_3 + 4p_4 + 5p_5$$

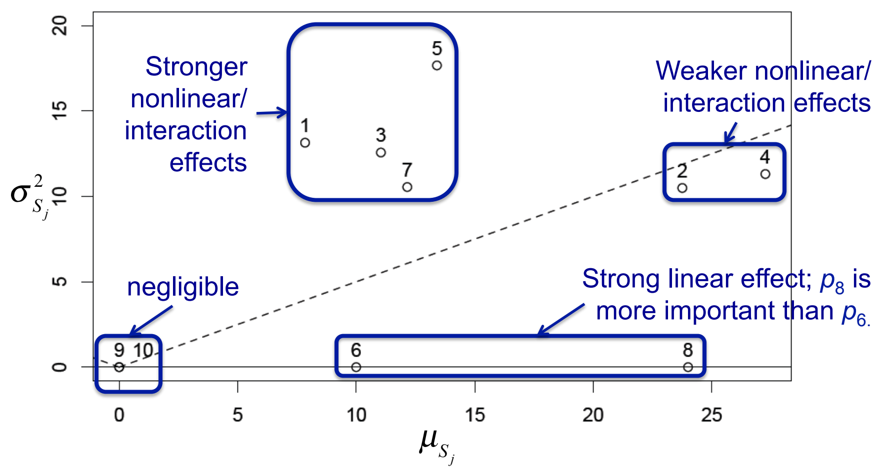
	p_1	p_2	p_3	p_4	p_5
μ_{S_j}	0.46	0.43	3.0	4.0	5.0
$\sigma_{S_j}^2$	0.11	0.10	0.0	0.0	0.0

- Parameters p_1 and p_2 show interaction:
  - The mean elementary effects are different from local sensitivity coefficients
  - Variances are non-zero
  - Cannot distinguish between nonlinearity and interaction!

20

MOAT

$$z(\mathbf{p}) = \sum_{i=1}^{10} \beta_i \omega_i + \sum_{i<j}^{10} \beta_{i,j} \omega_i \omega_j, \omega_i = p_i - 1, i = 2,4,6,8,9,10; \omega_i = p_i / (p_1 + 0.1) - 0.5, i = 1,3,5,7$$



21

Example: darcy6i

```
********
 OUTPUT
********
```
Look for `OUTPUT` in darcy6i.out

[… Outputs before the MOAT results give the samples and the corresponding objective function values for the paths taken to compute MOAT results …]

```
 MORRIS ONE-AT-A-TIME SENSITIVITY ANALYSIS
 =========================================
 Number of active parameters :    3
 Number of partitions        :    6
 Number of paths             :   12
 Total number of simulations :   48
```
Summary of parameters used in the MOAT

```
 Mean Elementary Effect for Each Observation
 -------------------------------------------

IOBS OBSERVATION   TIME      WEIGHT |                     log(abs. perm.) |
                                    |  MEAN EFFECT  MEAN ABS(EE)   STD. DEV. |
   4 Pressure 1/2  0.50 0.50000E-02  -0.14682E+01  0.74054E+01  0.86684E+01
   5 Flow inlet    0.50 0.12000E+05  -0.53067E+02  0.53067E+02  0.39115E+02
```

[Repeated for all observations …]

[Repeated for other parameters]

22

Example: darcy6i

```
Mean Elementary Effect for System State and Objective Function
==============================================================

*******************************************************************************
                                   SYSTEM STATE                                |
PARAMETER          ELEMENTARY EFFECT       MEAN ABS. EE.            STD. DEV.  |
-------------------------------------------------------------------------------
log(abs. perm.)         -0.10326E+02         0.21035E+02         0.26985E+02  |
Porosity                -0.20239E+01         0.24792E+01         0.42915E+01  |
Initial gas sat         -0.14251E+02         0.14823E+02         0.15299E+02  |
*******************************************************************************

[ … continue … ]

*************************************************************
             OBJECTIVE FUNCTION
   ELEMENTARY EFFECT       MEAN ABS. EE.           STD. DEV.
-------------------------------------------------------------
         0.12865E+05         0.13648E+05         0.23769E+05
         0.20162E+04         0.22682E+04         0.35059E+04
         0.51308E+04         0.51308E+04         0.58485E+04
*************************************************************
```

23

Variance-Based Global Sensitivity Analysis

- Basic concept: *Variance decomposition*
  - Which part of the output variance can be explained by which parameter?
  - How much can the output variance be reduced by fixing a parameter?
  - Requires Monte Carlo *sampling* → expensive!

24

Scatter Plots and Conditional Variances

- Perform Monte Carlo simulations, sampling input parameters (p_1,p_2,p_3,p_4)
- Plot $z(p_1,p_2,p_3,p_4)$ as a function of p_i
- Cut scatter plot *i* into thin vertical slices and take expected value, $E(z|p_i)$
- Take variance of $E(z|p_i)$, $V(E(z|p_i))$, to identify influential parameter

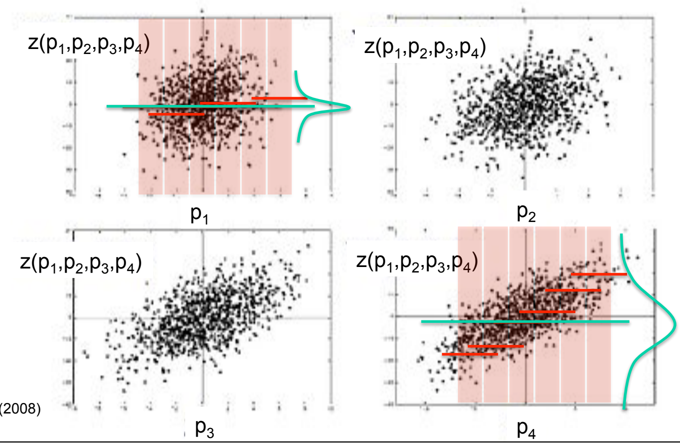


Figure after Saltelli et al. (2008)

25

Scatter Plots and Conditional Variances

- Perform Monte Carlo simulations
- Plot $Y(Z_1,Z_2,Z_3,Z_4)$ as a function of Z_i
- Cut scatter plot *i* into thin vertical slices and take expected value, $E(Y|Z_i)$
- Take variance of $E(Y|Z_i)$, $V(E(Y|Z_i))$, to identify influential parameter

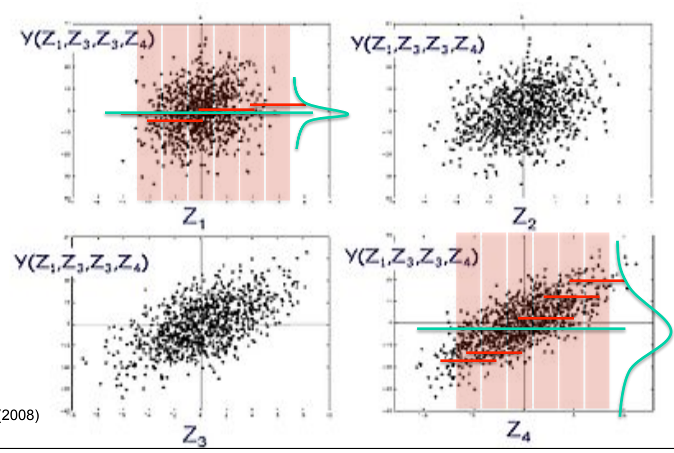


Figure after Saltelli et al. (2008)

26

Variance-Based Global Sensitivity Analysis

- Evaluate conditional variances
  - Fix one parameter, vary all the other $\mathbf{V}[\mathbf{E}_j(z_i \mid p_j)]$
    - first-order effect
    - no interaction
    - identifies influential parameters
    - Sobol' index
  - Vary one parameter, fix all the others $\mathbf{V}[\mathbf{E}_{-j}(z_i \mid p_{-j})]$
    - total effect
    - includes interaction effects
    - identifies non-influential parameters
    - total sensitivity index
- Number of forward simulations required = $N_{MC}(2+n)$

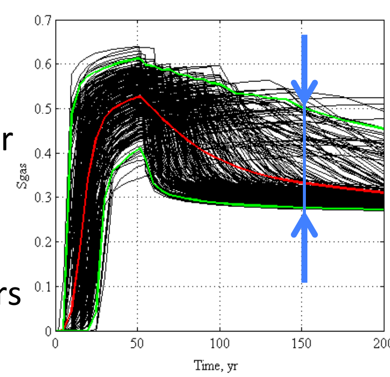
% contribution to uncertainty → sensitivity in UQ context

27

Sobol' Index

- Sobol' index
  - Fix one parameter, vary all the others
  - Measures variability with respect to individual parameter
  - First-order effect
  - Does not include interactions
  - Identifies influential parameters

$$S_i = \frac{V[E[Z \mid P_i]]}{V[Z]}$$



[Sobol, 2001; Saltelli, 2003]

28

Total Sensitivity Index

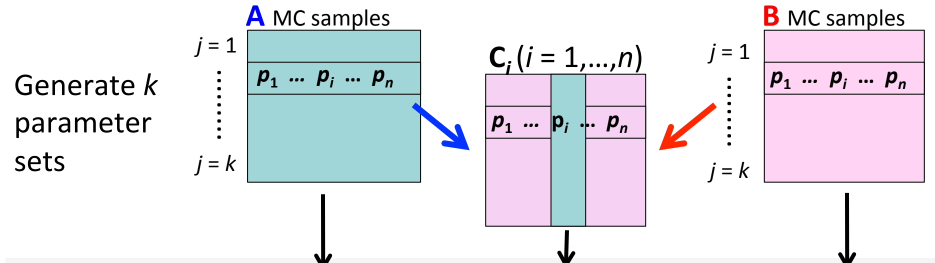
- Total sensitivity index
  - Vary one parameter, fix all the others
  - Total effect
  - Includes interactions
  - Identifies non-influential factors

$$S_{Ti} = 1 - \frac{V_{-i}[E[Z \mid P_{-i}]]}{V[Z]}$$

29

Compute Sobol' Indices

Saltelli (2003), Glen and Issacs (2011)



Compute $z(\mathbf{p})$ $\{a_1, a_2, \ldots, a_k\}$ $\{c_{i,1}, c_{i,2}, \ldots, c_{i,k}\}$ $\{b_1, b_2, \ldots, b_k\}$

Sobol' Index: S_i

$$S_i = \frac{1}{\sigma_Z^2}\frac{1}{k-1}\sum_{m=1}^{k}(a_m - m_Z)(c_{i,m} - m_Z)$$

→ Cor. ($\{a_m\}$, $\{c_{i,m}\}$)

Total Sensitivity Index: S_{ti}

$$S_{ti} = 1 - \frac{1}{\sigma_Z^2}\frac{1}{k-1}\sum_{m=1}^{k}(b_m - m_Z)(c_{i,m} - m_Z)$$

→ 1– Cor. ($\{b_m\}$, $\{c_{i,m}\}$)

30

Example: darcy7i

```
********
 OUTPUT
********
```

Look for `OUTPUT` in darcy7i.out

[… Outputs before the Saltelli Sensitivity Analysis results show samples and the corresponding objective function values used to evaluate sensitivity indices …]

```
SALTELLI SENSITIVITY ANALYSIS
=============================
Number of active parameters :    3
Number of samples           :  100
Total number of simulations :  500

Mean Elementary Effect for Each Observation
-------------------------------------------

IOBS OBSERVATION   TIME      WEIGHT |         log(abs. perm.)          |
                                    |   SENSITIVITY      TOTAL SENS.   |
   4  Pressure 1/2 0.50 0.50000E-02     0.70345E+00      0.10271E+01
   5  Flow inlet   0.50 0.12000E+05     0.49954E+00      0.11516E+01
```

Summary of parameters used in the Saltelli SA

[Repeated for all observations …]

[Repeated for other parameters]

Example: darcy7i

```
Mean Sensitivity for System State and Objective Function
========================================================

*********************************************************************************
                         SYSTEM STATE                  |      OBJECTIVE FUNCTION
PARAMETER        SENSITIVITY  TOTAL SENSITIVITY  |  SENSITIVITY  TOTAL SENSITIVITY
---------------------------------------------------------------------------------
log(abs. perm.) 0.53626E+00          0.94076E+00  |  0.28151E+00          0.10237E+01
Porosity        0.15732E+00          0.29462E+00  | -0.15934E-01         -0.72189E+00
Initial gas sat 0.22226E+00          0.37329E+00  |  0.25482E-01         -0.63849E+00
*********************************************************************************
```

Variance-based to Difference-based

Total sensitivity Index

$$S_{ti} = 1 - \frac{1}{\sigma_Z^2}\frac{1}{k-1}\sum_{m=1}^{k}(b_m - \mu_Z)\,(c_{i,m} - \mu_Z)$$

Sobol' Index

$$S_i = \frac{1}{\sigma_Z^2}\frac{1}{k-1}\sum_{m=1}^{k}(a_m - \mu_Z)\,(c_{i,m} - \mu_Z)$$

Covariance-semivariogram Relationship

$$S_{ti} = \frac{1}{\sigma_Z^2}\frac{1}{2(k-1)}\sum_{m=1}^{k}(b_m - c_{i,m})^2$$

Sobol' (2001)

- Perturb **one** parameter
- Take a difference

→ ~ Morris Mean EE²
Difference: σ_Z^{-2}, fixed pertub.

$$S_i = 1 - \frac{1}{\sigma_Z^2}\frac{1}{2(k-1)}\sum_{m=1}^{k}(a_m - c_{i,m})^2$$

- Perturb **other** parameters
- Take a difference

1 – total effects of others
= First-order effect

33

Recommendation

- Local SA should be done first
- Number of partitions in the Morris method
  → Small is acceptable (as long as the discrete points capture the variability)
- Examine the scatter plots from the Morris sampling
  → Nonlinear effects and interactions
- Sobol' method requires a large number of simulations
  → Confidence intervals
- Sobol' index does not differentiate minor parameters
  → Total sensitivity index (or Morris mean |*EE*|).
- MC simulations from UA
  → Sobol' index (approximation) as a by-product

34

Summary Comments on Sensitivity Analysis

- Perform Sensitivity Analysis as part of *test design*, i.e., *before* data collection
- Large sensitivity coefficients are a *necessary*, but *not sufficient* condition for inverse modeling
- Supplement Sensitivity Analysis with *synthetic inversions* for improved test design
- Maximize model *relevance*, *R*:

$$R = \frac{\text{number of parameters that induce significant variations in output of interest}}{\text{total number of model parameters}}$$

- *Model parsimony* may need to be assessed differently for diagnostic, inverse, and prognostic models

35

Related Topics

- Experimental Design
- Factor Prioritization / Parameter Screening
- Model Simplification / Factor Fixing / Reduced-Order Modeling
- Monte Carlo Filtering / Factor Mapping

References

Morris, M.D. (1991), Factorial Sampling Plans for Preliminary Computational Experiments, Technometrics, 33(2), 161-174.

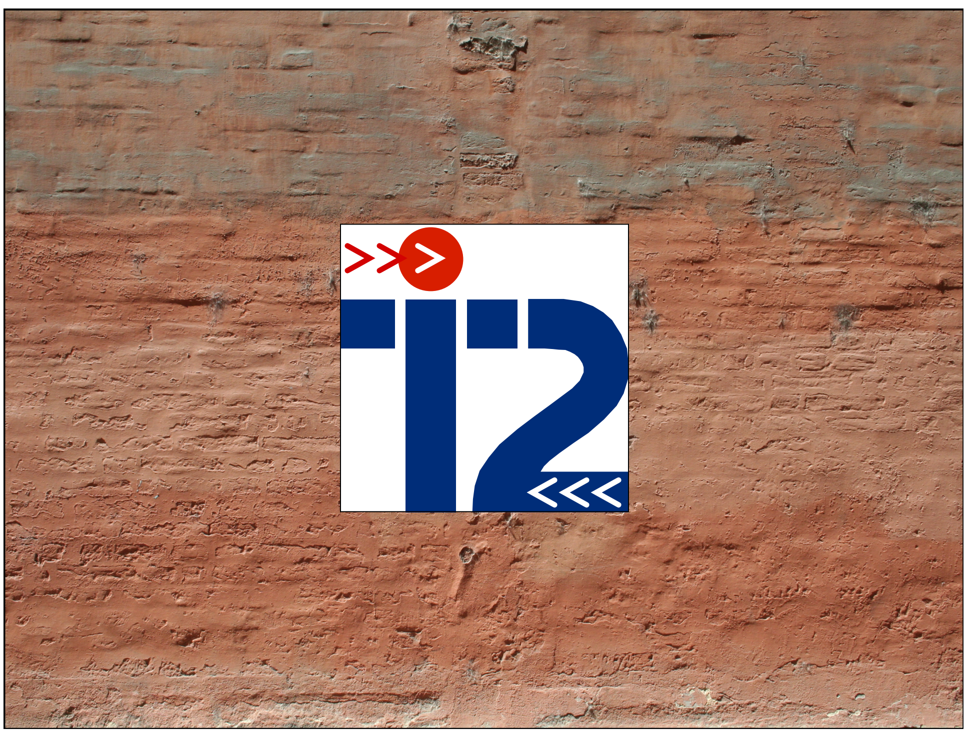
Saltelli et al.(2008), Global Sensitivity Analysis, The Primer, Wiley & Sons

36

Sensitivity Analysis: Questions

1. Purpose of sensitivity analysis?
2. Composite sensitivity measures?
3. Differences between local and global sensitivity analysis?
4. What do the mean and variance of the elementary effect tell you?
5. Discuss strengths and weaknesses of local and global sensitivity analysis methods

37



iTOUGH2 Commands
Local Sensitivity Analysis

```
> COMPUTATION
  >> OPTIONS
     >>> SENSITIVITY ANALYSIS
     <<<

  >> OUTPUT
     >>> print JACOBIAN for each iteration
     >>> print (unscaled) SENSITVITY matrix
     <<<
  <<
```

39

iTOUGH2 Commands
Numerical Evaluation of Derivatives

```
> COMPUTATION
  >> JACOBIAN
     >>> FORWARD : 3
```

Number of forward differencing before switching to centered differencing.

$$J_{ij} \approx \frac{z_i(\mathbf{p}; p_j + \delta p_j) - z_i(\mathbf{p})}{\delta p_j}$$

```
     >>> CENTERED
```

$$J_{ij} \approx \frac{z_i(\mathbf{p}; p_j + \delta p_j) - z_i(\mathbf{p}; p_j - \delta p_j)}{2\delta p_j}$$

More accurate but more expensive.

```
     >>> HESSIAN
```

$$\mathbf{H}_k = 2(\mathbf{J}_k^T \mathbf{C}_{zz}^{-1} \mathbf{J}_k + \mathbf{B})$$

Compute full finite-difference Hessian.

```
     >>> PERTURB : 0.01
```

Fraction of parameter value.
(default: 0.01)
(if negative, use as absolute perturbation)

40

iTOUGH2 Commands
Morris Sensitivity Analysis

```
> COMPUTATION
  >> OPTIONS
     >>> SENSITIVITY ANALYSIS : MORRIS OAT
         >>>> PATHS : 12
         >>>> PARTITIONS : 6
         <<<<
     <<<
  <<
```

Number of paths, n_p

Number of partitions, k

41

iTOUGH2 Commands
Saltelli Sensitivity Analysis

```
> COMPUTATION
  >> OPTIONS
     >>> SALTELLI GLOBAL SENSITIVITY ANALYSIS
         >>>> SAMPLES : 10000
         <<<<
     <<<
  <<
<
```

Number of Monte Carlo samples, N_{MC}

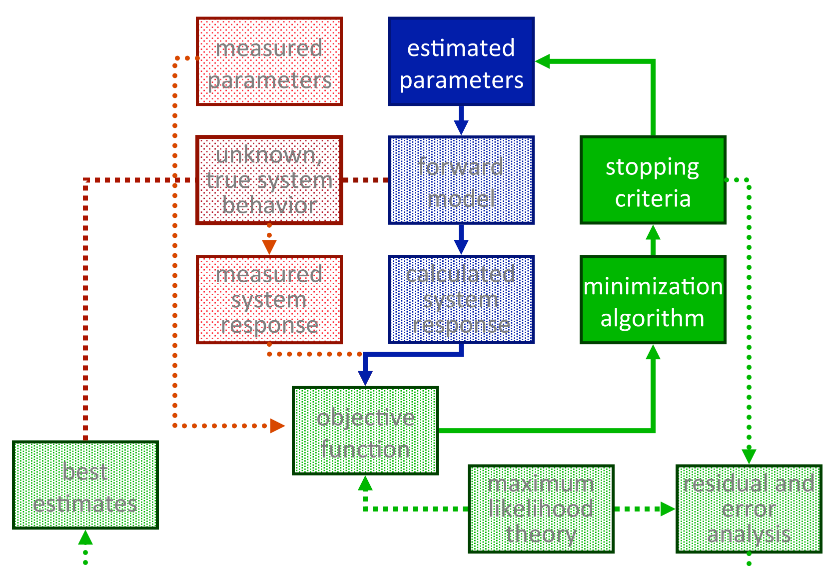
42

iTOUGH2 Short Course

Lawrence Berkeley National Laboratory
Earth Sciences Division
Berkeley, California

Minimization Algorithm

Overview and Classification
Gauss-Newton
Levenberg-Marquardt
Global Minimization Algorithms



2

Purpose of Minimization Algorithm

- *Find the minimum* of the objective function
- Automatically *update parameter vector* **p** such the objective function *S* is reduced.
  - Recall: The objective function *S* is a function of the model output **z**, which is a function of the parameter vector **p**.
  - For least-squares:

$$\textbf{minimize} \quad S = (\mathbf{z}^* - \mathbf{z}(\mathbf{p}))^T \mathbf{C}_{zz}^{-1} (\mathbf{z}^* - \mathbf{z}(\mathbf{p}))$$

3

Properties of Objective Function

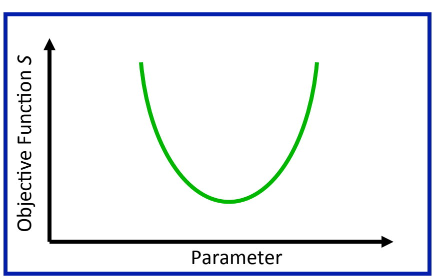
IDEAL

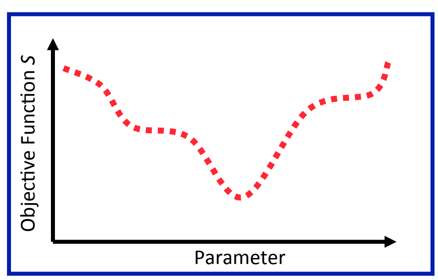
- Quadratic
- Symmetric and convex
- One global minimum
- Continuous
- Stable

approximate ?

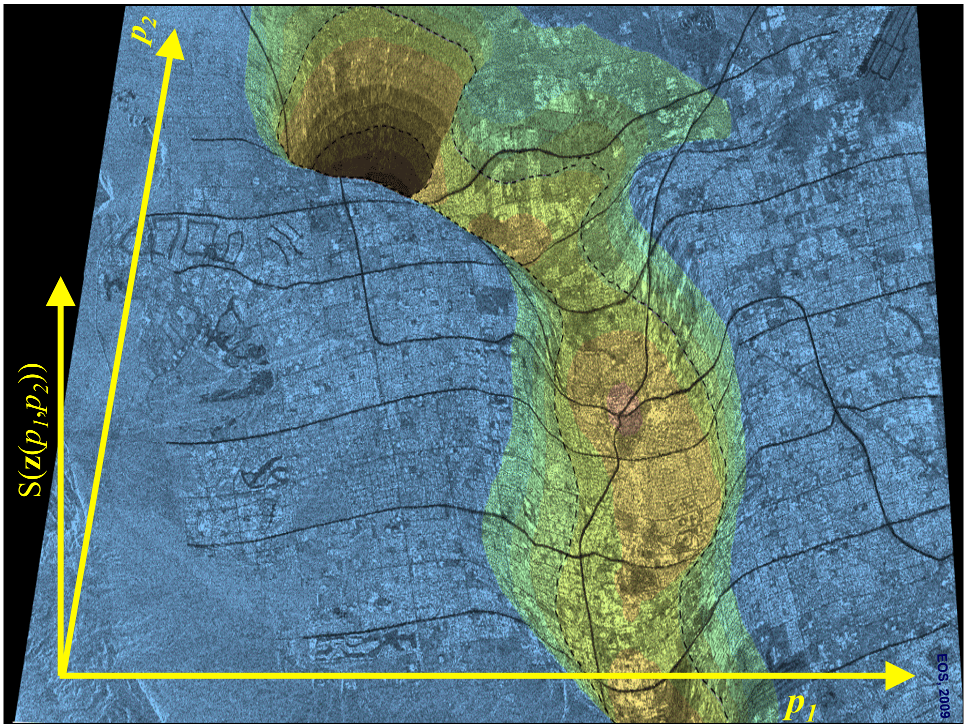
REAL

- Non-linear
- Complicated topology
- Many local minima
- Discontinuous
- Unstable





4



Steepest Descent

Analogy:

Skiing in fog

Algorithm:

Direction – Steepest descent

Step length – Until it goes back up

Problem: Local Minima

Classification of Algorithms

and

Direct Search Methods

Classification

- Global vs. Local
  – Find global minimum or nearest local minimum
- Direct Search Methods
  – Evaluate objective function many times
- Gradient-Based Methods
  – Move along gradient of objective function
- Second-Order Methods
  – Evaluate Hessian (or approximation to Hessian) of objective function

8

Direct Search Methods

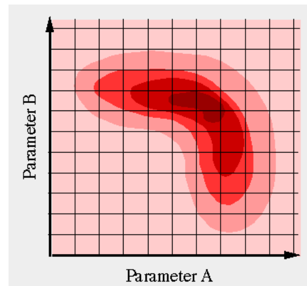
- *Principle*
  - Evaluate the objective function for systematically or randomly selected parameter combinations
- *Advantage*
  - No assumption about topology of objective function.
  - Obtain complete picture of parameter sensitivity and well- or ill-posedness of inverse problem
- *Disadvantage:* Inefficient
- *Examples*
  - Trial & Error
  - Grid Search
  - Sampling-based global methods

Trial & Error

- *Principle*
  - Update parameters based on expert's insight into system behavior and parameter sensitivities
- *Advantage*
  - Incorporation of "soft" information
  - Obtain feel for system behavior and sensitivities
- *Disadvantage*
  - Subjective
  - Tedious/inefficient (or impossible!)
  - No formal error analysis

Grid Search

- *Principle*
  - Evaluate objective function “everywhere” in the parameter space
- *Advantage*
  - Obtain complete information on:
    - Local minima
    - Sensitivities
    - Uncertainties
    - Nonuniqueness

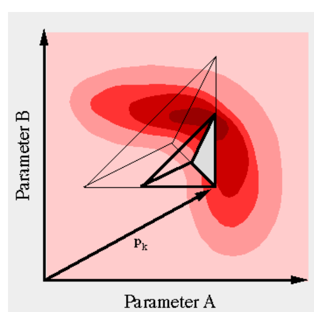


- *Disadvantage*
  - Very expensive: grows exponentially with *n*.
  - Only practical for up to n=3 parameters

11

Simplex Algorithm

- *Principle*
  - Obtain downhill direction from (*n*+1)-dimensional simplex. Move on by reflection, expansion, and contraction of simplex.
- *Advantage*
  - No derivatives
  - May jump over local minima
- *Disadvantage*
  - Relatively inefficient
  - No formal error analysis



12

Gradient-Based Methods

- *Principle*
  - Perform step along gradient direction
- *Advantage*
  - Robust for sufficiently small step sizes
  - There are efficient algorithms for calculating gradients (adjoints)
- *Disadvantage*
  - Inefficient stepping close to minimum
- *Examples*
  - Steepest descent
  - Quasi-Newton methods
  - Conjugate gradient methods

13

Second-Order Methods

- *Principle*
  - Evaluate second derivative of objective function or approximation thereof.
- *Advantage*
  - Quadratic convergence rate
- *Disadvantage*
  - Requires second derivatives
  - Not always robust
- *Examples*
  - Newton
  - Gauss-Newton
  - Levenberg-Marquardt

14

Gauss-Newton

and

Levenberg-Marquardt

Minimization Algorithms

Gauss-Newton Method

- Linearize model: $\mathbf{z} = \mathbf{z}_0 + \mathbf{J}\Delta\mathbf{p}$
- Substitute into objective function: $S = (\mathbf{z}^* - \mathbf{z}_0 - \mathbf{J}\Delta\mathbf{p})^T \mathbf{C}_{zz}^{-1} (\mathbf{z}^* - \mathbf{z}_0 - \mathbf{J}\Delta\mathbf{p})$
- Set derivative of objective function to 0: $\dfrac{\partial S}{\partial \mathbf{p}} = 0$
- Obtain solution $\Delta\mathbf{p}$: $\Delta\mathbf{p} = \left(\mathbf{J}^T \mathbf{C}_{zz}^{-1} \mathbf{J}\right)^{-1} \mathbf{J}^T \mathbf{C}_{zz}^{-1} \mathbf{r}$

The algorithm is iterative: Starting with k=0…

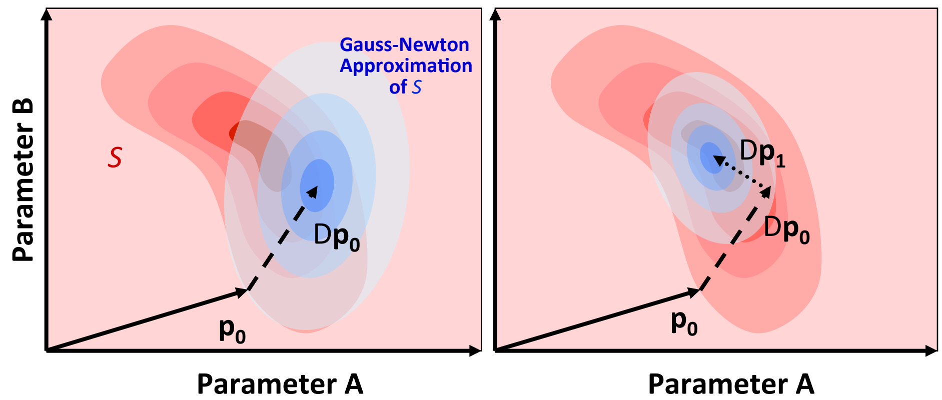
① Determine $\Delta\mathbf{p}_k$

② If $S(\mathbf{p}_k+\Delta\mathbf{p}_k) < S(\Delta\mathbf{p}_k)$, then set $\Delta\mathbf{p}_{k+1}=\mathbf{p}_k+\Delta\mathbf{p}_k$. Go to step 1.

③ If $S(\mathbf{p}_k+\Delta\mathbf{p}_k) > S(\Delta\mathbf{p}_k)$, then stop.

16

Gauss-Newton Method



17

Gauss-Newton Method

- Gauss-Newton identifies the minimum in a *single iteration* if:
  - Model is *linear*
  - Quadratic objective function
- Quadratic convergence rate for weakly nonlinear models
- Too large steps for strongly nonlinear models

18

Levenberg-Marquardt Method

- Modification of Gauss-Newton method for strongly nonlinear models
- Mixes gradient and Gauss-Newton method:
  - Performs robust steps along gradient far away from the minimum
  - Performs efficient Gauss-Newton steps near the minimum
- Automatically adjust relative weight between gradient and Gauss-Newton strategy

19

Levenberg-Marquardt Algorithm

$$\Delta\mathbf{p}_k = \left(\mathbf{J}_k^T\mathbf{C}_{zz}^{-1}\mathbf{J}_k + \lambda_k\mathbf{D}_k\right)^{-1}\mathbf{J}_k^T\mathbf{C}_{zz}^{-1}\mathbf{r}_k$$

λ_k = Levenberg parameter

If step is successful, move toward Gauss-Newton strategy (reduce λ by η)

If step unsuccessful, move toward gradient strategy (increase λ by η)

η_k = Marquardt parameter

D_k = Tikhonov matrix

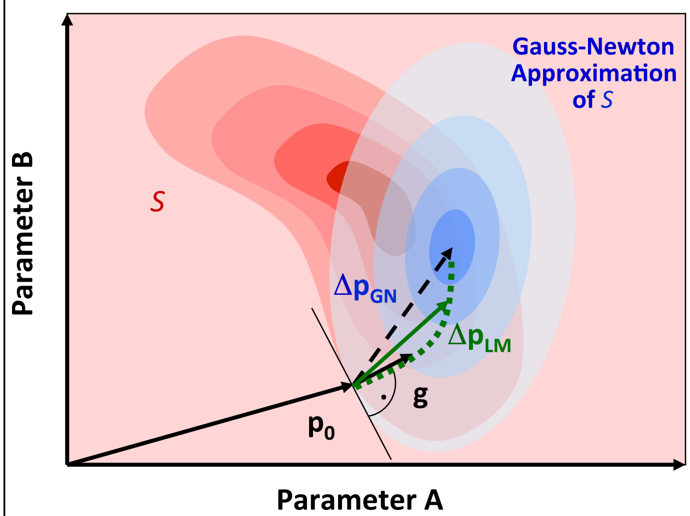
Identity matrix

Diagonal of normal matrix

Inverse of eigenvalues of normal matrix

20

Levenberg-Marquardt Method



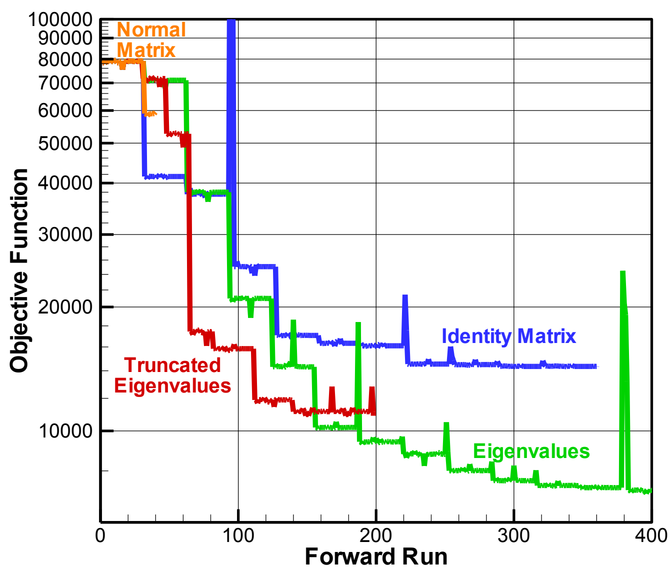
If stepping is successful, large Gauss-Newton steps are taken.

Otherwise, small, robust steps along steepest-descent direction are taken.

...... = possible endpoints of LM steps as a function of n

21

Levenberg-Marquardt Algorithm



22

Example: darcy3i

Iteration

Objective function

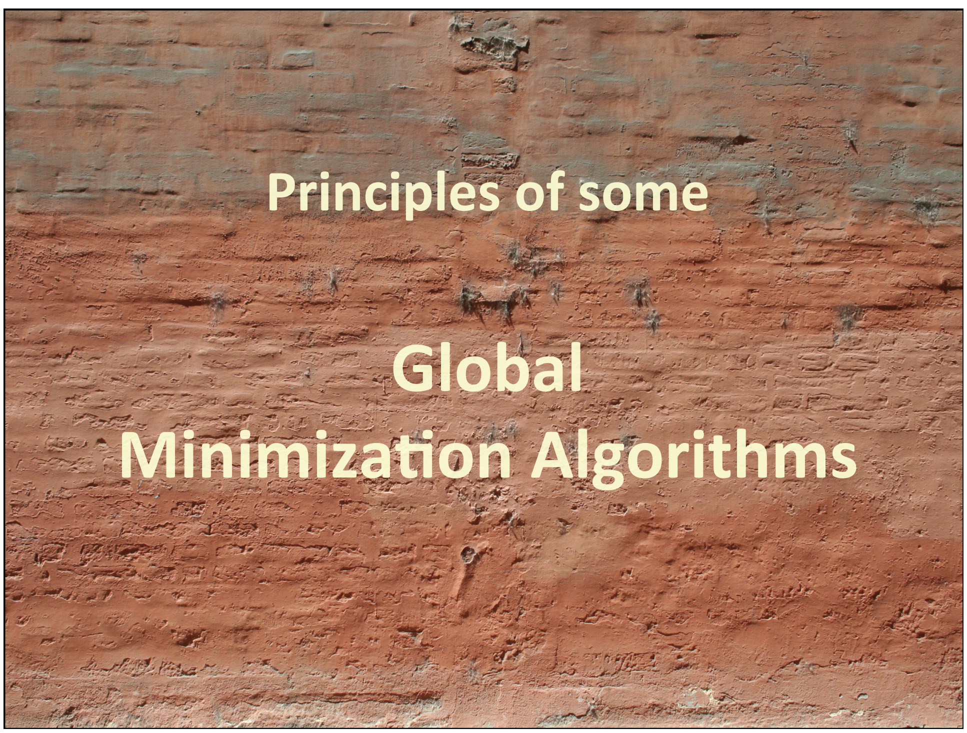
Scaled step size

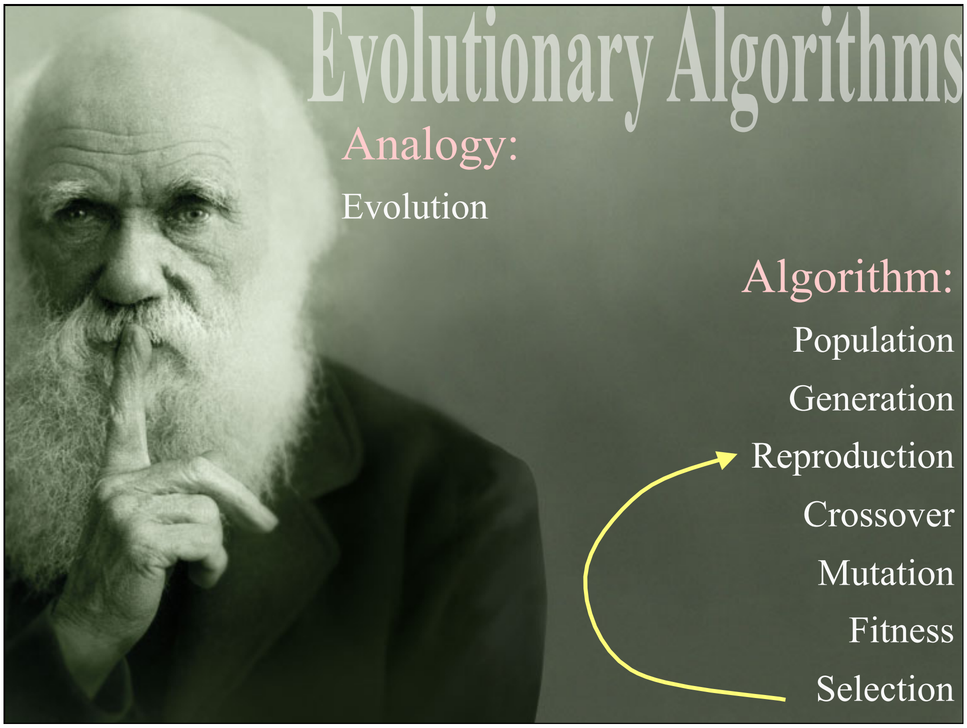
Parameter set

Parameter update

```
LEVENBERG-MARQUARDT ALGORITHM
--------------------------------------------------------------------------------
ITER TOUGH2 OBJ FUNC. MAX. RESID. EQU. log(abs. perm.) Porosity Initial gas sat
--------------------------------------------------------------------------------
>I   0     1 0.52083E+03 0.62277E+02    5  -0.120000E+02  0.250000E+00  0.102500E+02
 Finite-difference update of Jacobian matrix
 J   1 Gradient      =    0.76583E+04 (forward)
 ------------------------------------------------------------------
 S     Step size     =    0.73094E-01   Scaled step size =  0.401228E-01
 SVD   Singular values of (JT*P*J)   :       0.101128E+04  0.125868E+02  0.413918E+04
 PU    Log(LP)=   0  Parameter update:       0.709579E-01 -0.991490E-02  0.144712E-01
>I   1     5 0.29236E+03 0.47037E+02    5  -0.119290E+02  0.240085E+00  0.102645E+02
 Finite-difference update of Jacobian matrix
 J   2 Gradient      =    0.57722E+04 (forward)
[ ... Intermediate  Steps ...]
 ------------------------------------------------------------------
 S     Step size     =    0.57239E-06   Scaled step size =  0.208041E-05
 SVD   Singular values of (JT*P*J)   :       0.142812E+04  0.160688E+02  0.338576E+04
 PU    Log(LP)= -5  Parameter update:        0.177621E-06  0.520074E-06 -0.160002E-06
>I   7    35 0.43026E+02 0.52897E+01   28  -0.116867E+02  0.372566E+00  0.102915E+02
 Finite-difference update of Jacobian matrix
 J   8 Gradient      =    0.31979E+01 (centered)
 ------------------------------------------------------------------
 CS    Step size     =    0.19640E-07   Scaled step size =  0.702879E-07
    Step tolerance =  0.100000E-08   --> Terminate!
```

23



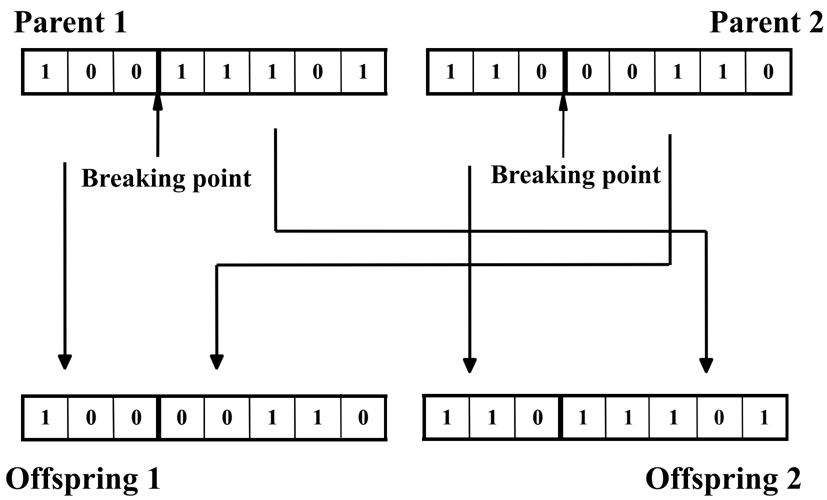


Analogy

- **Chromosome –** individual (a potential solution) made of arrays of genes (values of decision variables)
- **Populations –** a group of potential solutions
- **Generation –** result of a GA iteration
- **Fitness –** a relative measure of individual quality (objective function)
- **Selection –** a process to select individuals that are more likely to reproduce
- **Crossover –** mixing genes from two parents
- **Mutation –** a random alteration of individual (explore new search area)

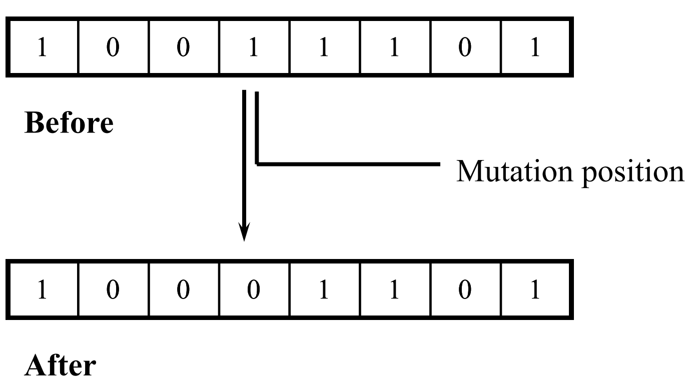
26

Crossover

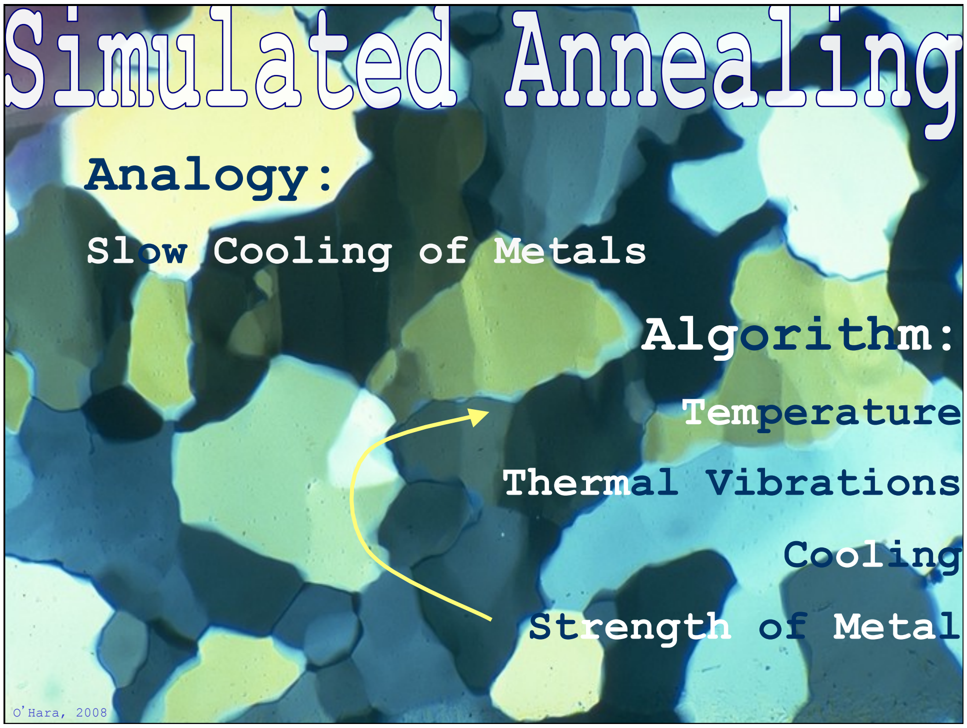


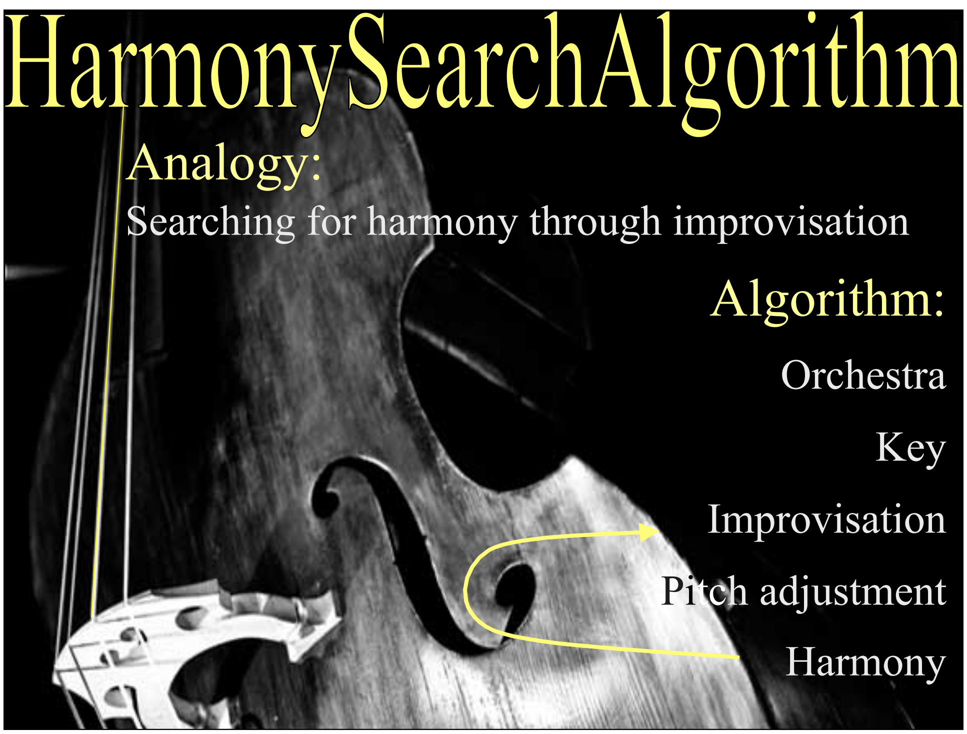
27

Mutation

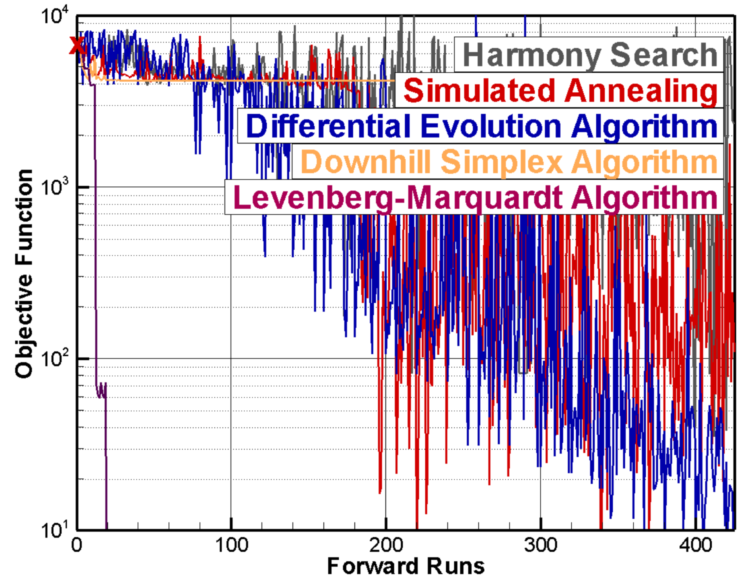


28

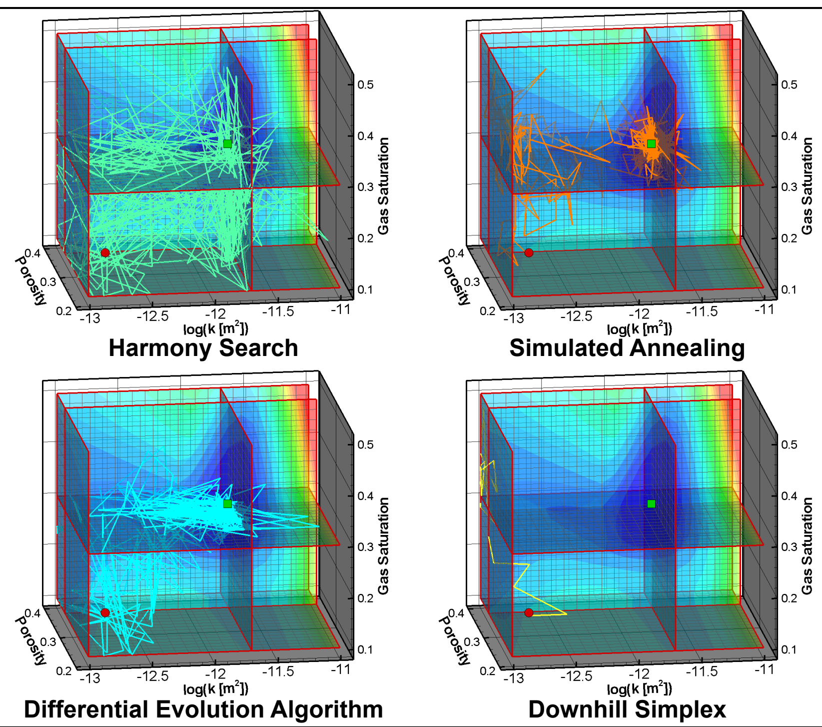

Simulated Annealing
Analogy:
Slow Cooling of Metals
Algorithm:
Temperature
Thermal Vibrations
Cooling
Strength of Metal
O'Hara, 2008


HarmonySearchAlgorithm
Analogy:
Searching for harmony through improvisation
Algorithm:
Orchestra
Key
Improvisation
Pitch adjustment
Harmony

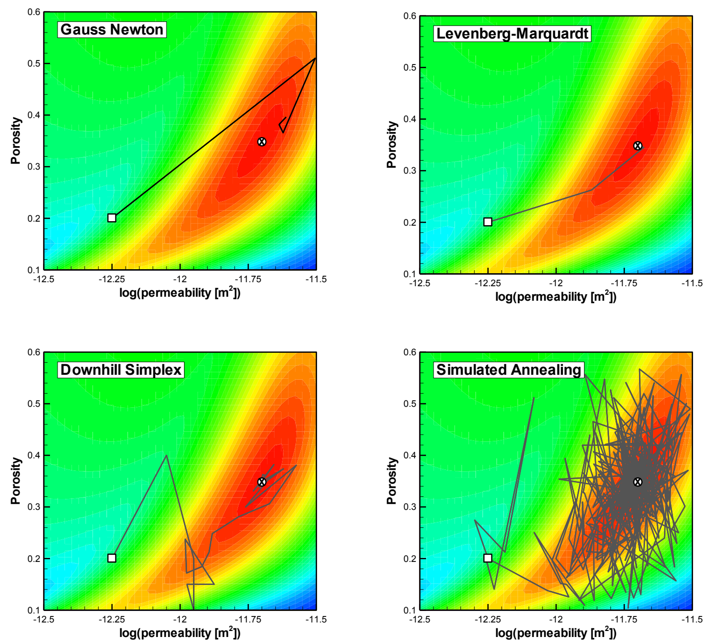
Performance of Global Minimization Algorithms



31



32



33

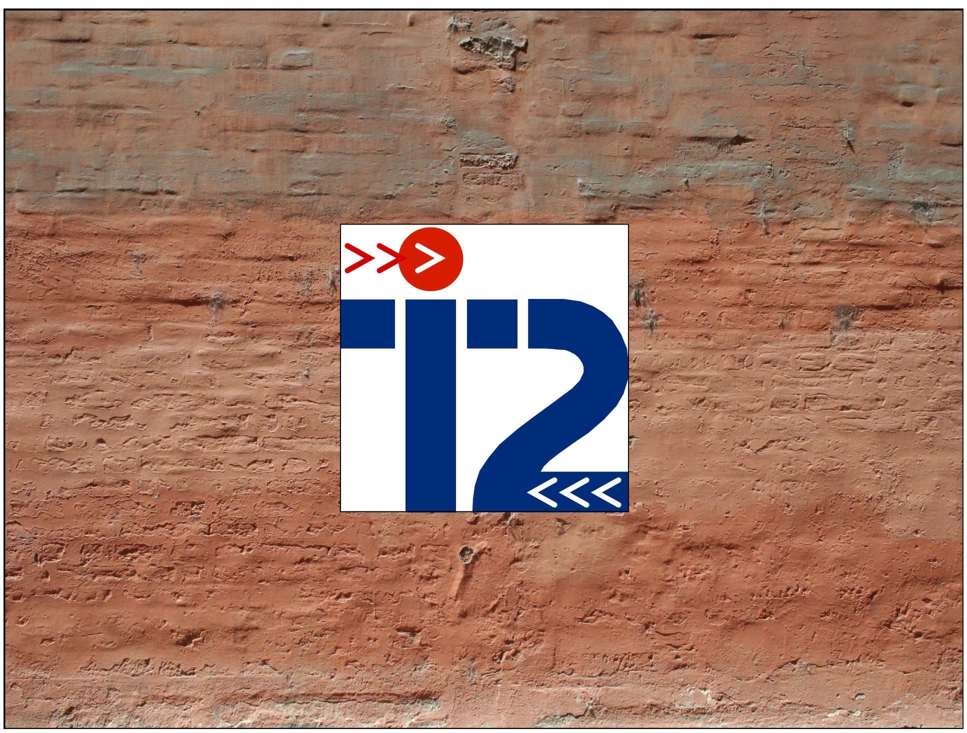
Common Features of Global Algorithms

- Deterministic step based on objective function
- Random component to escape local minima
- Heuristic control parameters
- Inefficient

Minimization Algorithm: Questions

1. Purpose of minimization algorithm?
2. List methods and their advantages and limitations
3. Key criteria for selecting the minimization method?
4. Sketch:
   - Objective function for two parameters, nonlinear model
   - Select starting point (initial guess of parameter vector)
   - Draw Gauss-Newton approximation
   - Perform two Gauss-Newton steps
5. Principle of Levenberg-Marquardt method?

35



iTOUGH2 parameters and observations

Check all second-level commands in `> PARAMETER` block for list of adjustable parameters that can be estimated.

Check all second-level commands in `> OBSERVATION` block for list of observations that can be used for model calibration.

Code unsupported, user-specified parameters and observations into subroutines `USERPAR` and `USEROBS` (file *it2user.f*), respectively.

Use PEST protocol to include parameters of pre-processors or observations calculated by a post-processor.

37

> PARAMETER Commands Affecting Minimization Algorithms

```
> PARAMETER
  >> any
     >>> any
         >>>> initial GUESS: par0
         >>>> RANGE: min max
         >>>> PERTURB: (-)pert (for Jacobian)
         >>>> maximum STEP size per iteration
         >>>> VALUE/LOGARITHM/FACTOR/LOG(F)
         >>>> DEVIATION: sigma
     <<<
```

38

Local Minimization Algorithms

```
> COMPUTATION
  >> STOP
     >>> ITERATIONS: 10
     >>> max. scaled STEP size: 10.0
     <<<

  >> alternative OPTIONS
     >>> FORWARD run
     >>> LEVENBERG-MARQUARDT
     >>> GAUSS-NEWTON
     >>> DOWNHILL SIMPLEX

  >> JACOBIAN
     >>> PERTURB: 0.03
     >>> CENTERED
```

39

Levenberg-Marquardt Algorithm

```
> COMPUTATION
  >> STOP
     >>> ITERATION  : 10
     >>> LEVENBERG  : 10.0
     >>> MARQUARDT  : 10.0
     >>> STEP       :  1.0
     >>> UPHILL     :    5
     >>> STAY ALIVE
     >>> NO ABORT
     <<<
  >> alternative OPTIONS
     >>> LEVENBERG-MARQUARDT EIGENVALUE
     >>> LEVENBERG-MARQUARDT IDENTITY
     >>> LEVENBERG-MARQUARDT SVD TRUNCATE
```

Initial Levenberg parameter λ. Default is 1.0.

A larger value leads to a smaller, safer initial step.

If λ=0, the algorithm is equivalent to Gauss Newton.

Maximum scaled step size. Default is unbounded.

Maximum number of unsuccessful uphill steps.

40

Grid Search and Monte Carlo

```
> COMPUTATION
 >> OPTION
    >>> GRID SEARCH: 9 9 19
```

```
> COMPUTATION
  >> STOP
     >>> do: 1000 MC SIMULATIONS
     >>> allow runs to ABORT as soon as OF>Ofmin
     >>> resample to STAY ALIVE!!!
     <<<

  >> ERROR
     >>> use MONTE CARLO simulations and report run
         with minimum objective function value
     >>> LATIN HYPERCUBE SAMPLING
```

41

Global Minimization Algorithms: Harmony Search and Simulated Annealing

```
> COMPUTATION
  >> OPTIONS
     >>> HARMONY search algorithm
         >>>> MEMORY size: 20
         >>>> CONSIDERATION: 0.8
         >>>> PITCH adjustment: 0.3
         <<<<

     >>> SIMULATED ANNEALING
         >>>> initial TEMPERATURE: -0.03
         >>>> update after a maximum of : 20 STEPS
         >>>> annealing SCHEDULE: 0.95
         >>>> maximum number of ITERATIONS: 100
         <<<<
```

42

Global Minimization Algorithms: Differential Evolutionary Algorithm

```
> COMPUTATION
  >> OPTION
     >>> DIFFERENTIAL EVOLUTION ALGORITHM
         >>>> number of POPULATIONs     : 20
         >>>> MUTATION scaling factor   : 0.8
         >>>> CROSSOVER scaling factor  : 0.8
         >>>> MUTOPTION                 : 2
         >>>> RANDOM COMBINED FACTOR
         >>>> STRATEGY                  : 6
         <<<<
```

43

Minimization Algorithms: >> OUTPUT Commands

```
> COMPUTATION
  >> OUTPUT
     >>> print OBJECTIVE FUNCTION for each run
     >>> print RESIDUALS for each run
     >>> print JACOBIAN after each iteration
     >>> create NEW OUTPUT file for each run
     >>> PLOTTING curves after each: iteration
     >>>
```

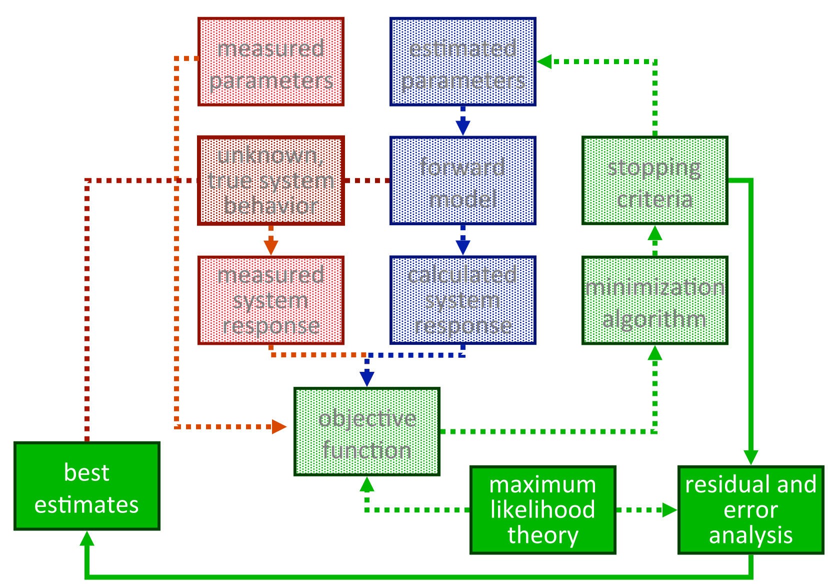
44

iTOUGH2 Short Course

Lawrence Berkeley National Laboratory
Earth Sciences Division
Berkeley, California

Residual Analysis

Efficiency Criteria
Fisher Model Test
Residual Analysis



2

Why Residual and Uncertainty Analysis?

- Parameter estimates may be worthless if:
  - Model with optimized parameter does not match the data, i.e., it is an unlikely representation of the true system
    goodness-of-fit, Fisher Model Test
  - Estimates are biased by systematic errors or outliers in the data
    residual analysis
  - Estimation uncertainty is large
    $\mathbf{C}_{pp}$, *correlation coefficients*
  - Solution is *non-unique* or *unstable*

3

Residual Analysis

- Types of Residuals
- Residual Analysis
- Overall Measures of Misfit
  - Estimated Error Variance
  - Fisher Model Test
  - Nash-Sutcliffe Efficiency
  - Kling-Gupta Efficiency

Type of Residuals: Systematic

- Systematic error in data:
  - Data conversion error: factor, shift, drift
  - Experimental error: leak, forcing term
- Systematic error in model:
  - Inadequate process description
  - Inadequate parameterization
  - Inadequate model structure
- *Contain information* about parameters of interest
  - A function of parameters.
- *Inconsistency* between real system and its model representation
- If systematic residuals persist *after* inversion, adjust experiment *or* model!

5

Type of Residuals: Random

- Errors in stochastic model
  - Distributional assumption
  - Heteroscedasticity: non-uniform variance
  - Correlations
- Reducing errors in stochastic model
  - Analyze residuals
  - Use appropriate transformations
  - Include correlations
  - Use robust estimators

$$\tilde{z}(z;\lambda) = \begin{cases} \dfrac{(z^{\lambda}-1)}{\lambda \cdot g^{\lambda-1}} & \textbf{if } \lambda \neq 0 \\ g \cdot \ln(z) & \textbf{if } \lambda = 0 \end{cases}$$

$$\tilde{r}_i = \begin{cases} r_1\sqrt{(1-\rho^2)} & \textbf{if } i = 1 \\ r_i - \rho \cdot r_{i-1} & \textbf{if } i = 2,\ldots,m \end{cases}$$

$$\hat{r} = \begin{cases} 1-\cos(\tilde{r}/c) & \textbf{if } |\tilde{\mathbf{r}}| \leq \mathbf{c}\pi \\ 2 & \textbf{if } |\tilde{r}| > \mathbf{c}\pi \end{cases}$$

6

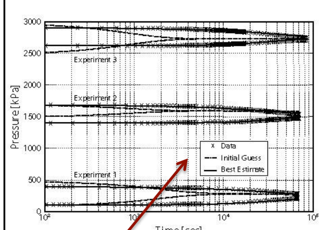
Steps of Residual Analysis

1. Plot the residual
   → look for randomness
2. Standard statistics (mean, median, skewness, kurtosis)
   → look for symmetry
3. Regression analysis on measure vs. calculated
   → look for zero intercept, unit slope, and unit Pearson's *R* (coefficient of determination)
4. Fisher, Nash-Sutcliffe and Kling-Gupta efficiency criteria
   → close to 1
5. Runs statistics
   → Statistically tests number of positive and negative runs in residuals

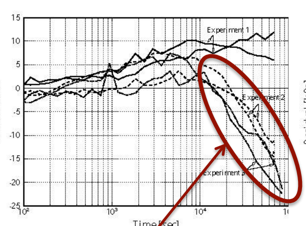
7

Remedies

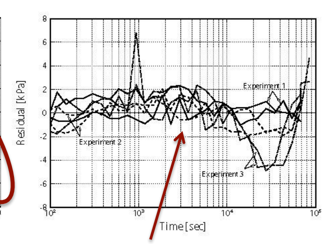
- Residuals should be *random*
- Trends and patterns in residuals indicate *systematic errors*
- Try to *remove systematic errors* by:
  - Refining the functional model
  - Correcting systematic data errors
  - Parameterizing and estimating conceptual model elements
  - Parameterizing and estimating measurement errors



Perfect match between data and model



Systematic errors from gas leakage



Random residuals after gas leakage is explicitly modeled

8

Covariance Matrices

- *A priori* error variance (assumed 1.0): σ_0^2
- *A priori* covariance matrix of observations: $\mathbf{C}_{zz} = \sigma_0^2 \cdot \mathbf{V}_{zz}$
- *A posteriori* error variance: $s_0^2 = \dfrac{\mathbf{r}^T \mathbf{C}_{zz}^{-1} \mathbf{r}}{m-n}$
- Covariance matrix of estimated parameters:

$$\mathbf{C}_{pp} = s_0^2 \left(\mathbf{J}^T \mathbf{C}_{zz}^{-1} \mathbf{J} \right)^{-1}$$

- Covariance matrix of calculated observation: $\mathbf{C}_{\hat{z}\hat{z}} = \mathbf{J}^T \mathbf{C}_{pp} \mathbf{J}$
- Covariance matrix of residuals: $\mathbf{C}_{rr} = \mathbf{C}_{zz} - \mathbf{C}_{\hat{z}\hat{z}}$
- *i*th diagonal element of covariance matrix: $\sigma_{z_i}^2, \sigma_{\hat{z}_i}^2, \sigma_{r_i}^2$

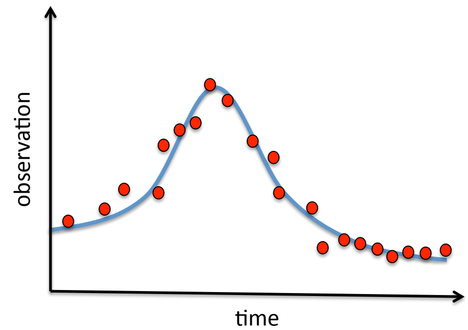
9

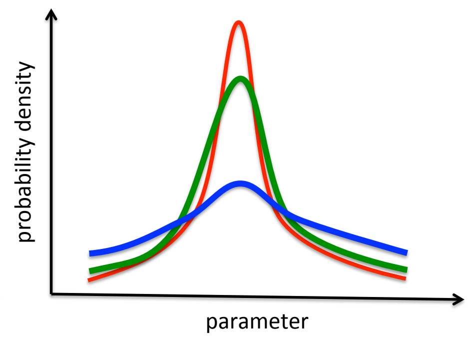
Residual Analysis Output in iTOUGH2

- The following are tabulated in the output file (*.out).
  - `RESIDUAL:` $r_i = z_i^* - z_i$
  - `C.O.F.:` Relative contribution [%] to objective function
  - `STD. DEV.:` *A posteriori* standard deviation of calculated system response $\mathbf{C}_{\hat{z}\hat{z}} = \mathbf{J}^T \mathbf{C}_{pp} \mathbf{J}$
  - `Yi:` Local reliability or influence ➔ $y_i = 1 - \left(\sigma_{\hat{z}_i} / \sigma_{z_i} \right)^2$
    Observations with $y_i < 0.25$ are poorly controlled
  - `Wi:` Studentized residual ➔ $w_i = r_i / \sigma_{r_i}$
    Potential outlier if $\mathrm{abs}(w_i) > u(0.95) = 1.96$
  - `IOD:` Relative impact [%] of omitting observation based on D-optimality criterion ("data-worth analysis")

10

Evaluate the worth of data points
- without having data!





$$\text{rel. data worth} = \frac{\det(\mathbf{C}_{pp}) - \det(\mathbf{c}_{pp})}{\det(\mathbf{C}_{pp})}$$

$$\mathbf{C}_{pp} = s_0^2\left(\mathbf{J}^T\mathbf{C}_{zz}^{-1}\mathbf{J}\right)^{-1}$$

$$\mathbf{c}_{pp} = s_0^2\left(\mathbf{j}^T\mathbf{c}_{zz}^{-1}\mathbf{j}\right)^{-1}$$

Example: darcy3i

```
********************
 RESIDUAL ANALYSIS
********************
```

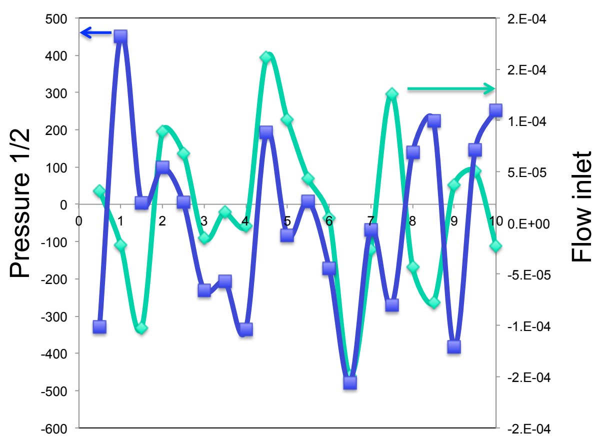
[… Description for the columns of the table …]

```
-------------------------------------------------------------------------
   #  OBSERVATION   AT TIME [min]     MEASURED      COMPUTED      RESIDUAL
-------------------------------------------------------------------------
   1  log(abs. perm.)             -0.12000E+02 -0.11687E+02 -0.31333E+00 }
   2  Porosity                     0.25000E+00  0.37257E+00 -0.12257E+00 } prior info
   3  Initial gas sat              0.10250E+02  0.10291E+02 -0.41477E-01 }
   4  Pressure 1/2    0.50000E+00  0.10370E+06  0.10367E+06  0.35837E+02
   6  Pressure 1/2    0.10000E+01  0.10317E+06  0.10327E+06 -0.10851E+03  [cont...]
   8  Pressure 1/2    0.15000E+01  0.10257E+06  0.10291E+06 -0.33246E+03
```

[… Repeated for all observations …]

12

Example: darcy3i



Appear random, with no clear trend in the data

Summary Statistics

- `MEAN`: mean of the residual (should be close to zero)
- `STD. DEV.`: standard deviation (should be consistent with stochastic model, i.e., *a priori* defined standard deviations)
- `M/S`: Ratio of mean and standard deviation; indicates whether mean (bias) is significant (should be small)
- `SKEWNESS`: Degree of asymmetry of residuals (should be 0)
- `KURTOSIS`: Relative peakedness of distribution (should be 0)

$$SKEW = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{r_i - \bar{r}}{SDEV}\right)^3 \qquad KURT = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{r_i - \bar{r}}{SDEV}\right)^4 - 3$$

Note: statistical measures may not be robust for small data sets

Summary Statistics

- `MEDIAN`: median of the residuals (should be close to 0)

$$\hat{r} = \begin{cases} r_{(m+1)/2} & \text{odd } m \\ 0.5 r_{m/2} + r_{m/2+1} & \text{even } m \end{cases}$$

- `AVE. DEV.`: Mean absolute deviation (should be close to 0)

$$ADEV = \frac{1}{m}\sum_{i=1}^{m} \left| r_i - \hat{r} \right|$$

- Large differences between median and mean and between standard deviation and average deviation indicate robustness issue

15

Example: darcy3i

```
Summary of Residual Analysis
----------------------------
Max weighted residual at observation    :            28
Max weighted residual                   :    -0.2300E+01
Max residual                            :    -0.4600E+03
Number of poorly controlled observations:             0
Number of large normalized residuals    :             3
Max normalized residual at observation  :            28
Max normalized residual                 :          2.40
Probable size of maximum error          :     0.5021E+03
```

[... Some iteration statistics ...]

```
Control Measures
----------------
Trace (P*QLL) : n   =   3               :     0.3000E+01
Sum   (Yi)    : m-n =  37               :     0.3700E+02

Objective Function                                          C.O.F.
------------------
Initial value of objective function     :     0.5208E+03    1210.52 %
Minimum value of objective function     :     0.4303E+02     100.00 %
```

16

Example: darcy3i

[… Summary statistics for each dataset …]

```
=========================================================================
DATASET                    DATAPOINTS           MEAN        MEDIAN     STD. DEV.
-------------------------------------------------------------------------
Pressure 1/2        [Pa]        20     -0.137E+02    -0.287E+02     0.210E+03
Flow inlet          [kg/sec]    20     -0.708E-06     0.628E-05     0.898E-04
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
ALL RESIDUALS       WEIGHTED    43     -0.359E-01    -0.000E+00     0.101E+01
=========================================================================
```

Consistent with measurement noise added to data in *nonoise.dat* to create *noise.dat*

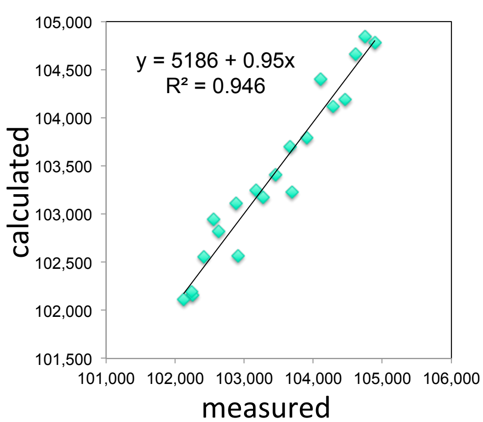
[… Summary statistics for each datatype …]

```
=========================================================================
DATATYPE                   DATAPOINTS           MEAN        MEDIAN     STD. DEV.
-------------------------------------------------------------------------
PRESSURE            [Pa]        20     -0.137E+02    -0.287E+02     0.210E+03
FLOW RATE           [kg/sec]    20     -0.708E-06     0.628E-05     0.898E-04
=========================================================================
```

17

Linear Regression Analysis

- `INTERCEPT` (should be 0)
- `SLOPE` (should be 1)
- `R`: coefficient of determination (should be 1)
- `NSE`: Nash-Sutcliffe Efficiency (should be 1)
- `KGE`: Kling-Gupta Efficiency (should be 1)
- `GAMMAi`: Contribution of correlation error, variability error and bias error to the total error



18

Nash-Sutcliffe Efficiency (*NSE*)

- A normalized model performance criterion, with observed mean as baseline

$$NSE = 1 - \frac{\sum_{i=1}^{m}(z_i - z_i^*)^2}{\sum_{i=1}^{m}(z_i^* - \mu_{obs})^2} = 1 - \frac{(\mathbf{r}^T\mathbf{r})/m}{\sigma_{obs}^2}$$

- Popular criterion in hydrology
- *NSE* ≤ 1; ideal value: *NSE* = 1
  → *NSE* can be used as objective function to be maximized
- If *NSE* ≤ 0, the model is not a better predictor than using the observed mean
- Decomposition: *NSE* = *A* – *B* – *C*

$A = R^2$: Correlation between *z*(**p**) and *z**

$B = [R - (\sigma_{calc} / \sigma_{obs})]^2$: Variability

$C = [(\mu_{calc} - \mu_{obs}) / \sigma_{obs}]^2$: Bias

Nash-Sutcliffe Efficiency (*NSE*)

- Decomposition helps examine the different components of the residuals
- Murphy (1988) : $NSE = A - B - C$

$A = R^2$: Correlation between *z*(**p**) and *z**

$B = [R - (\sigma_{calc} / \sigma_{obs})]^2$: Conditional bias

$C = [(\mu_{calc} - \mu_{obs}) / \sigma_{obs}]^2$: Unconditional bias

- Gupta (2009) : $NSE = 2\alpha R - \alpha^2 - \beta^2$

R : Correlation between *z*(**p**) and *z**

$\alpha = (\sigma_{calc} / \sigma_{obs})$: Relative variability

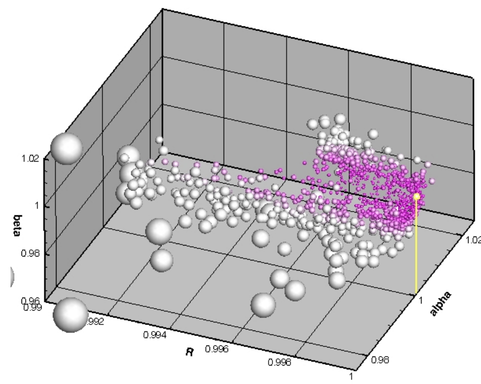
$\beta = (\mu_{calc} / \mu_{obs})$: Relative bias

Kling-Gupta Efficiency (*KGE*)

- Reweights different components derived from the decomposed *NSE*

$$KGE = 1 - \sqrt{(R-1)^2 + (\alpha - 1)^2 + (\beta - 1)^2}$$

- Measures Euclidian distance from ideal point:
  $R = 1, \quad \alpha = 1, \quad \beta = 1$
- *KGE* ≤ 1
  ideal value: *KGE* = 1
  → *KGE* can be used as a multi-objective calibration criterion to be maximized



Kling-Gupta Efficiency (*KGE*)

- Relative contribution of the 3 components:

$$\Gamma_i = \frac{G_i}{G_1 + G_2 + G_3}$$ where $$G_1 = (R-1)^2$$ $$G_2 = (\alpha - 1)^2$$ $$G_3 = (\beta - 1)^2$$

- For more information:
  Gupta et al., *Journal of Hydrology*, *377*, 80 - 91, 2009

Example: darcy3i

```
Linear Regression Analysis of Calculated Versus Observed Data ...
-------------------------------------------------------------

=============================================================
DATASET                DATAPOINTS      INTERCEPT        SLOPE
-------------------------------------------------------------
Pressure 1/2       [Pa]       20      0.506E+04    0.951E+00
Flow inlet         [kg/sec]   20     -0.884E-05    0.985E+00
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
ALL                WEIGHTED   40      0.366E+00    0.100E+01
=============================================================


====================================================================
       R       NSE          KGE         GAMMA1       GAMMA2       GAMMA3
--------------------------------------------------------------------
0.973        0.943        0.964        0.590        0.410        0.000
0.869        0.755        0.824        0.554        0.446        0.000
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
1.000        1.000        1.000        0.000        0.512        0.488
===================================================+================
```

23

Estimated Error Variance s_0^2

- The *estimated error variance*
  - a posteriori error variance
  - an aggregate measure of *goodness-of-fit*
  - represents the *mean weighted residual*

$$s_0^2 = \frac{\mathbf{r}^T \mathbf{C}_{zz}^{-1} \mathbf{r}}{m - n}$$

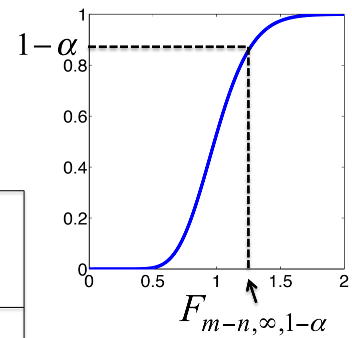
r : residual
$\mathbf{C}_{zz}$: observation covariance matrix
m : number of observations
n : number of parameters

- If *a priori* assumptions about the residuals (expressed through matrix $\mathbf{C}_{zz}$) were reasonable, then s_0^2 / σ_0^2 is close to 1: Fisher Model Test
- Recall: Solving a weighted least-squares problem minimizes the estimated error variance

24

Fisher Model Test

- The ratio s_0^2 / σ_0^2 is a random variable following an *F*-distribution
- $F_{m-n,\infty,1-\alpha}$ is the inverse of the cumulative *F*-distribution for 1-α



$s_0^2 / \sigma_0^2 > F_{m-n,\infty,1-\alpha}$ Error in functional or stochastic model
$1 \le s_0^2 / \sigma_0^2 \le F_{m-n,\infty,1-\alpha}$ Model test passed!
$s_0^2 / \sigma_0^2 \le 1$ Error in stochastic model or "overfitting"

- Only meaningful if a reliable stochastic model is available

25

Example: darcy3i vs. darcy4i

```
Fisher Model Test
-----------------
Root mean square error               : 0.1078E+01
Estimated error variance             : 0.1163E+01
Variance used for error analysis     : 0.1187E+01  (a posteriori variance)
Nash-Sutcliffe efficiency criterion  :     1.0000
Degree of freedom                    :         37  (no prior information)
Confidence level (1-alpha)           :       95.0  [%]
Lucky you                            : Model test successful!
                                       --> Estimated error variance is used!

Fisher model test criterion          :       1.41  (F-distribution)
Factor for confidence bands          :       2.03  (t-distribution)
Factor for confidence regions        :       2.91  (chi-square distribution)
```

noisy.dat

```
Root mean square error               : 0.3183E+01
Estimated error variance             : 0.1013E+02
Variance used for error analysis     : 0.1013E+02  (a posteriori variance)
Nash-Sutcliffe efficiency criterion  :     0.9999
Degree of freedom                    :         37  (no prior information)
Confidence level (1-alpha)           :       95.0  [%]
Warning                              : Model test failed!
                                       --> Estimated error variance is used!

Fisher model test criterion          :       1.41  (F-distribution)
Factor for confidence bands          :       2.03  (t-distribution)
Factor for confidence regions        :       2.91  (chi-square distribution)
```

noisier.dat

26

Model Identification Criteria

- Criteria to allow comparison of different inversions
  - Different data sets, parameters, or conceptual models
- Goodness of fit (s_0)
  - Not appropriate since adding more parameters always improves the fit (risk of overparameterization)
- Optimality criteria: A: trace($\mathbf{C}_{pp}$) E: max eig($\mathbf{C}_{pp}$) D: det($\mathbf{C}_{pp}$)
- Akaike Information Criterion

$$AIC = (m-n)s_0^2 + \ln|\mathbf{C}_{zz}| + m\ln(2\pi) + 2n$$

- Akaike's Bayesian Information Criterion

$$AIC = (m-n)s_0^2 + \ln|\mathbf{C}_{zz}| + m\ln(2\pi) + \ln|\mathbf{J}^T\mathbf{C}_{zz}^{-1}\mathbf{J}|$$

- Kashyap Criterion

$$d_k^* = (m-n)s_0^2 + \ln|\mathbf{C}_{zz}| + m\ln(2\pi) + n\ln(m/2\pi) + \ln|\mathbf{J}^T\mathbf{C}_{zz}^{-1}\mathbf{J}|$$

Example: darcy3/4i

$$c_{ij}^* = \frac{c_{ij}}{p_i p_j}$$

```
Optimality Criteria  [noisy.dat]                  unscaled        scaled
-------------------
D-optimality = det(Cpp)                :       0.2622E-12     0.1306E-15
A-optimality = trace(Cpp)              :       0.5905E-03     0.2072E-02
E-optimality = max eigenvalue          :       0.5323E-03     0.2129E+00
Log-likelihood ln(L)                   :       0.2362E+02
Akaike  =-2ln(L)+2n                    :      -0.4123E+02
ABIC    =-2ln(L)+ln|F|                 :      -0.2168E+02
Kashyap =-2ln(L)+ln|F|+n*ln(m/2Pi)     :      -0.1591E+02
```

negative log-likelihood to be minimized

```
Optimality Criteria [noisier.dat]                 unscaled        scaled
-------------------
D-optimality = det(Cpp)                :       0.1493E-09     0.1139E-12
A-optimality = trace(Cpp)              :       0.4288E-02     0.1812E-01
E-optimality = max eigenvalue          :       0.3789E-02     0.1601E+00
Log-likelihood ln(L)                   :      -0.1423E+03
Akaike  =-2ln(L)+2n                    :       0.2906E+03
ABIC    =-2ln(L)+ln|F|                 :       0.3081E+03
Kashyap =-2ln(L)+ln|F|+n*ln(m/2Pi)     :       0.3139E+03
```

All criteria have lower values for noisy.dat because of higher data quality, making it the preferred model

Questions Residual Analysis

1. The *a posteriori* error variance s_0^2 turns out to be significantly greater than the *a priori* error variance σ_0^2.
   - What does "significantly" mean?
   - What does that result indicate?

2. What are you looking for when evaluating residuals?

29

iTOUGH2 Short Course

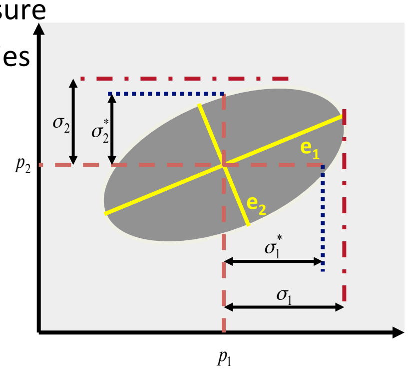
Lawrence Berkeley National Laboratory
Earth Sciences Division
Berkeley, California

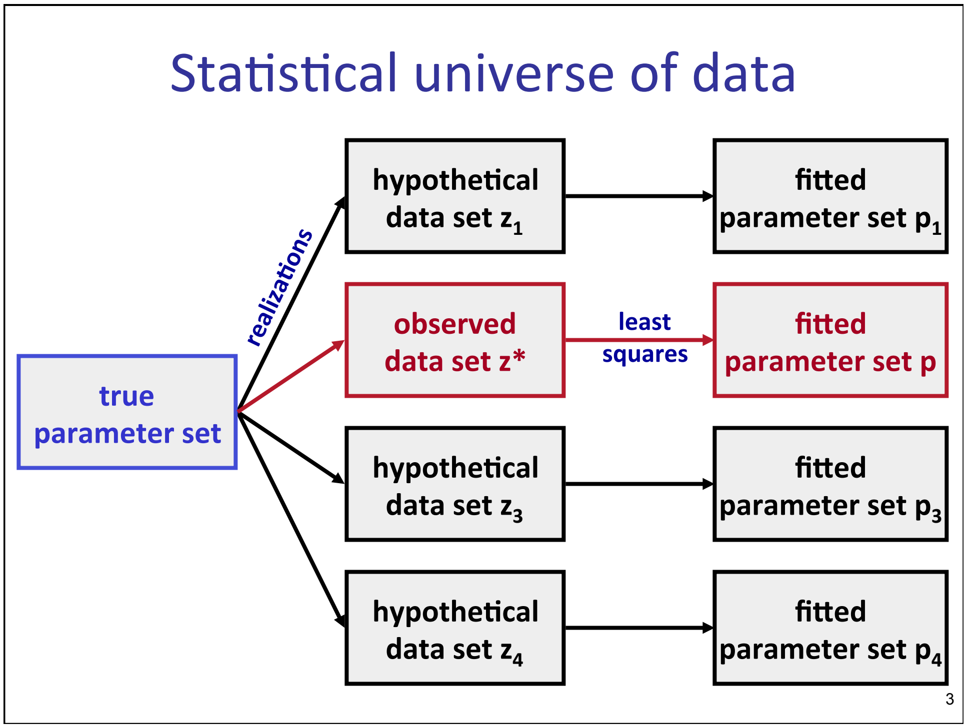
Uncertainty Analysis

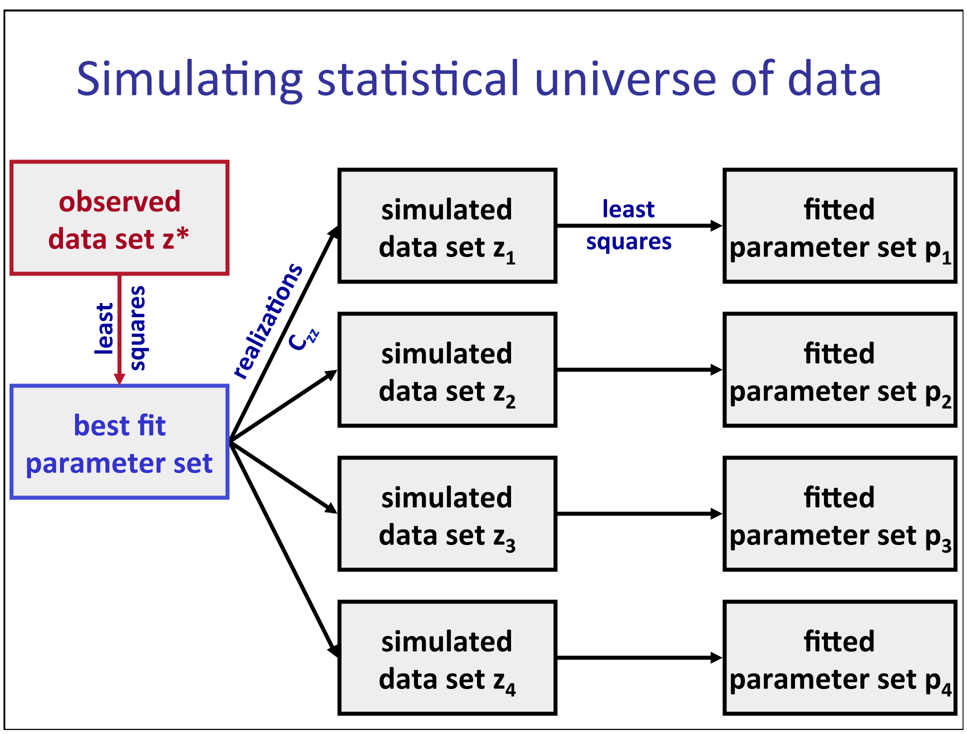
Covariance Matrix of Estimated Parameters C_{pp}
Uncertainty Propagation Analysis

Parameter uncertainty analysis in iTOUGH2

- Covariance/correlation matrix
- Correlation chart
- Direct (pairwise) parameter correlations
- Conditional estimation uncertainties
- Overall parameter correlation measure
- Improvement over prior uncertainties
- Eigenanalysis of covariance matrix
- Correction for nonlinearities



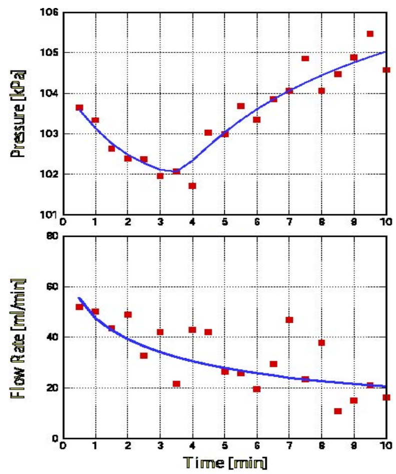

Statistical universe of data
true parameter set
realizations
hypothetical data set z_1
fitted parameter set p_1
observed data set z*
least squares
fitted parameter set p
hypothetical data set z_3
fitted parameter set p_3
hypothetical data set z_4
fitted parameter set p_4
3

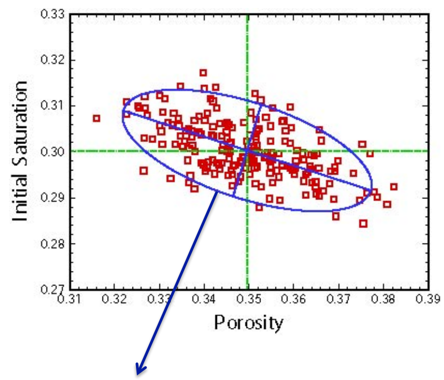

Simulating statistical universe of data
observed data set z*
least squares
best fit parameter set
realizations
C_{zz}
simulated data set z_1
least squares
fitted parameter set p_1
simulated data set z_2
fitted parameter set p_2
simulated data set z_3
fitted parameter set p_3
simulated data set z_4
fitted parameter set p_4

Simulating estimation uncertainty

200 realizations of hypothetical pressure and flow rate data

200 best estimates of porosity and initial gas saturation





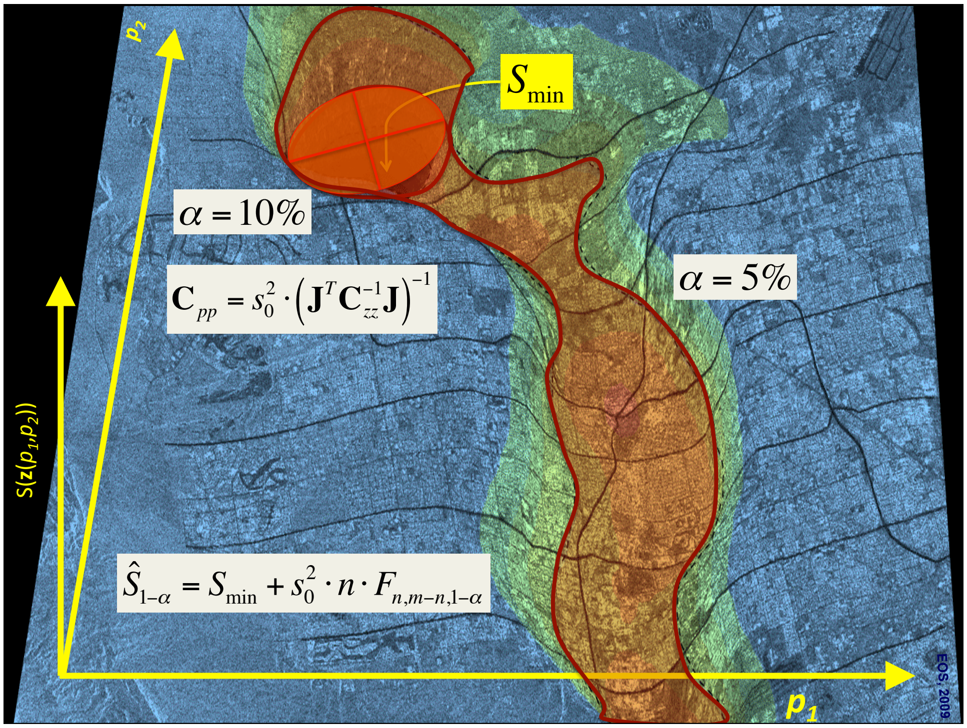
$\mathbf{C}_{pp}$ gives reasonable estimate of uncertainty region

5

Confidence Region

- The contours of the objective function visualize:
  - Confidence region
  - Correlation structure
  - Appropriateness of linearity assumption
  - Appropriateness of normality assumption
  - Well-posedness of inverse problem

6



Confidence Region

- The probability that the true parameter set lies within the ellipsoidal confidence region represented by $\mathbf{C}_{pp}$ is (1-α).
- The true confidence region is bounded by the contour line of the objective function at level:

$$S_{\min} + s_0^2 \cdot n \cdot F_{m-n,\infty,1-\alpha}$$

- The confidence region increases with decreasing α.
- $\mathbf{C}_{pp}$ is only a good representation of actual confidence region if linearity and normality assumptions are not violated.

8

Covariance Matrix of Estimated Parameters $\mathbf{C}_{pp}$

- The covariance matrix $\mathbf{C}_{pp}$ is an estimate of the uncertainty of the estimated parameters:

$$\mathbf{C}_{pp} = s_0^2\left(\mathbf{J}^T\mathbf{C}_{zz}^{-1}\mathbf{J}\right)^{-1}$$

- $\mathbf{C}_{pp}$ is an approximation of the actual parameter uncertainty; it is based on a normality and linearity assumption
- $\mathbf{C}_{pp}$ is proportional to goodness-of-fit (s_0^2)
- $\mathbf{C}_{pp}$ is inversely proportional to sensitivity matrix (**J**)
- $\mathbf{C}_{pp}$ is proportional to measurement uncertainty ($\mathbf{C}_{zz}$)

9

Estimation Uncertainty

- Decreases with improvement of fit
  - *Use good data and good model*
- Decreases with increasing sensitivity
  - *Use sensitive data*
- Decreases with decreasing correlations
  - *Use data that allow for independent determination of each parameter*
  - *Avoid overparameterization*
- Design tests accordingly!

10

Covariance Matrix
Correlation Matrix

- The diagonal of $\mathbf{C}_{pp}$ contains variances σ_{ii}^2 of estimated parameters.
- Off-diagonal elements are covariances c_{ij} between pairs of parameters.
- Off-diagonal elements are "normalized" to yield *correlation coefficients* r_{ij}:

$$-1 < r_{ij} = \frac{c_{ij}}{\sqrt{\sigma_{ii}^2 \cdot \sigma_{jj}^2}} < 1$$

11

Correlation Coefficients

- A correlation coefficient of *zero* indicates that the two parameters can be estimated *independently*.
- A correlation coefficient of *–1 or 1* indicates *non-uniqueness*.
- A negative correlation coefficient indicates that a statistically similar match can be obtained by increasing one parameter and decreasing the other.
- If correlations exist, the uncertainty in one parameter affects the uncertainty in the other parameters.
- Design experiment as to minimize correlations.

12

Example: darcy3i

```
****************
 ERROR ANALYSIS
****************

Error analysis is based on >>> a posteriori <<< variance:  1.1628577E+00


Covariance(L+D)/Correlation(U) Matrix of Estimated Parameters
-------------------------------------------------------------
                   log(abs. perm.)        Porosity  Initial gas sat

log(abs. perm.)        2.77543E-04           0.843           -0.259
Porosity               2.35163E-04     2.80104E-04           -0.490
Initial gas sat       -1.74370E-05    -3.32081E-05      1.63676E-05


Matrix of Direct Correlations
-----------------------------
                   log(abs. perm.)        Porosity  Initial gas sat
log(abs. perm.)              1.000           0.851            0.331
Porosity                     0.851           1.000           -0.525
Initial gas sat              0.331          -0.525            1.000
```

Correlation coefficient, r_{ij}

σ_p^2

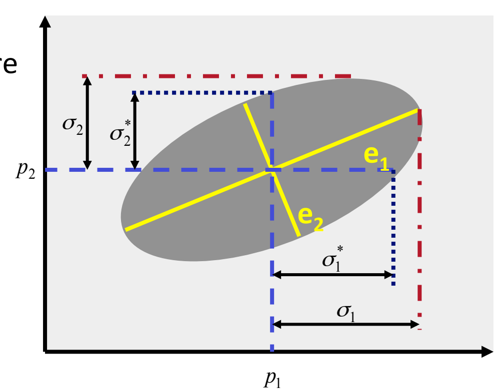
covariance, c_{ij}

Indicates degree to which a change in one parameter can be compensated by another

13

Overall Correlation

- The *conditional* standard deviation σ^* is the estimation uncertainty assuming that all other parameters are perfectly known.
- C_{pp} holds the *marginal* standard deviations σ.
- The ratio σ^*/σ is a measure of overall correlation.
- The ratio σ^*/σ should be close to 1.



14

Example: darcy3i

Arrange in order of decreasing independence (C/M)

Square root of the diagonal of C_{pp}

```
Standard Deviations
-------------------
[ ... description of the column ... ]

PARAMETER          BEST ESTIMATE  A PRIORI  CONDITIONAL   MARGINAL    C/M  1-M/P
Initial gas sat     1.02915E+01     N/A     3.32706E-03 4.04569E-03 0.822 1.000
log(abs. perm.)    -1.16866E+01     N/A     8.44619E-03 1.66596E-02 0.507 1.000
Porosity            3.72609E-01     N/A     7.65512E-03 1.67363E-02 0.457 1.000
```

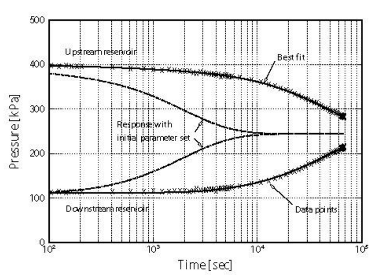
Specified only if used to weight parameters in objective function

Measure of how much is learned from inversion compared to a priori uncertainty

```
Correlation Chart
-----------------

-0.6     -0.4     -0.2    0.0                        0.0      0.2      0.4      0.6      0.8
|--------|--------|--------|        PARAMETER         |--------|--------|--------|--------|
      |---------|---------|      Initial gas sat
      |         |
      |         |---------|      log(abs. perm.)     |---------------------------------|
      |                                                                                  |
      |-------------------|         Porosity         |---------------------------------|
```

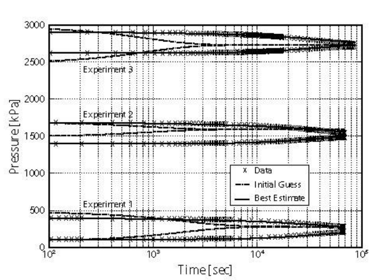
Correlation between gas saturation and porosity

Reducing Correlations: Example



	log(*k*)	log(*b*)	ϕ
log(*k*)	1.67	-0.99	-0.87
log(*b*)	-0.99	2.16	0.87
ϕ	-0.87	0.87	0.003

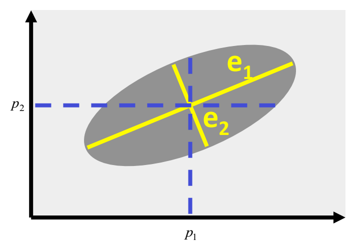
Add experiments on different pressure levels and invert jointly



	log(*k*)	log(*b*)	ϕ
log(*k*)	1E-4	-0.52	-0.12
log(*b*)	-0.52	4e-4	-0.02
Φ	-0.12	-0.02	0.01

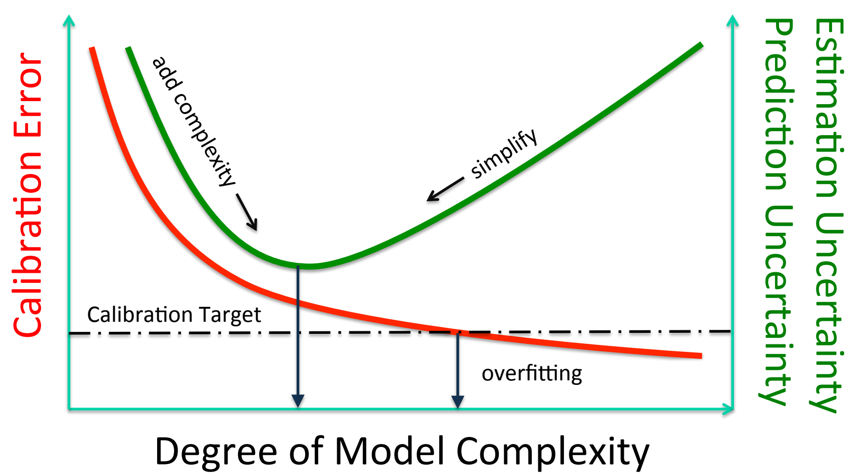
Example: darcy3i

```
Eigenanalysis of Covariance Matrix (R-Mode Factor Analysis)
-----------------------------------------------------------
Performance index      :  3.25365122E-02
Condition number       :  5.07978268E+01

Eigenvalues
-----------
                                      1               2               3
  1 Eigenvalue     :         4.7286641E-05   5.1655936E-04   1.0168926E-05

Eigenvectors
------------
                                      1               2               3
  1 log(abs. perm.)          6.9034965E-01   7.0218708E-01  -1.7421439E-01
  2 Porosity                -6.5282457E-01   7.0839235E-01   2.6832882E-01
  3 Initial gas sat          3.1182918E-01  -7.1509271E-02   9.4744340E-01
```



17

Under- and overparameterization: Occam's Razor



18

Overparameterization

- A match can *always* be improved by adding more parameters to **p**.
- Adding new parameters increases correlations and thus *increases estimation uncertainty*.
- Check $\mathbf{C}_{pp}$ for large variances, correlation coefficients close to –1 or 1, and large condition numbers.
- Add parameters only if the fit can be *significantly improved* without introducing strong parameter correlations.
- *Avoid over-* ***and*** *under-parameterization*

19

Uncertainty Analysis: Questions

1. Discuss

$$\mathbf{C}_{pp} = s_0^2\left(\mathbf{J}^T\mathbf{C}_{zz}^{-1}\mathbf{J}\right)^{-1}$$

2. Under which conditions is $\mathbf{C}_{pp}$ a good approximation of the actual confidence region?
3. How can you reduce estimation uncertainty?

20

Uncertainty Analysis: Questions

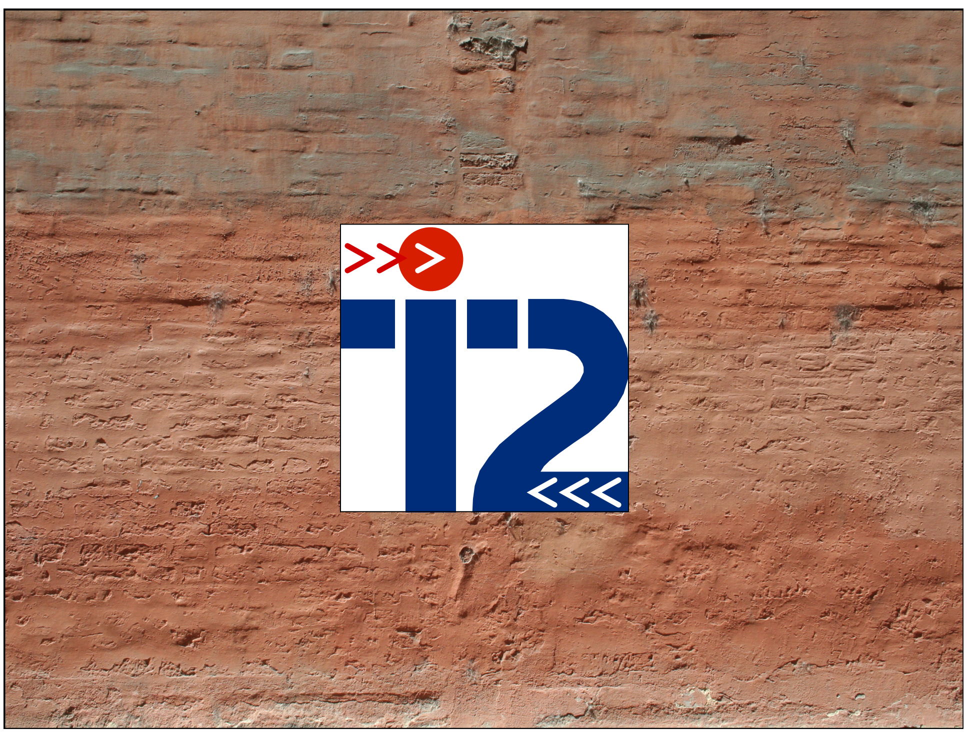
4. Discuss "underparameterization" and "overparameterization".

5. $\mathbf{C}_{zz} = \sigma_0^2\, \mathbf{V}_{zz}$ is the *a priori* observation covariance matrix.

   If all elements of $\mathbf{C}_{zz}$ were multiplied by a factor of 4, how would this affect:

   - The value of the objective function S?
   - The estimated parameter set $\mathbf{p}$?
   - The estimated error variance s_0^2?
   - The uncertainty of the estimated parameters $\mathbf{C}_{pp}$?
   - The outcome of the Fisher Model Test?

21



iTOUGH2 Commands

> COMPUTATION
>> ERROR
>>> ALPHA: *alpha* significance level
>>> A PRIORI use σ_0^2
>>> A POSTERIORI use s_0^2
>>> let FISHER model test decide
>>> check LINEARITY assumption

23

Prediction Uncertainty

- Uncertainty Propagation Analysis
  - Linear Uncertainty Propagation Analysis
  - Sampling-Based Uncertainty Quantification

Uncertainty Propagation Analysis

- Calculate prediction uncertainty as a result of parameter uncertainty.
- *Linear analysis* (First-Order Second-Moment)
  - Fast (n+1 forward runs)
  - Easy to report (mean and covariance matrix)
  - Based on linearity and normality assumption
- *Monte Carlo simulations*
  - Expensive (many forward runs)
  - Difficult to report
  - Full distribution
  - No distributional assumptions

25

Linear Error Propagation Analysis (First-Order-Second-Moment)

- *Assumptions*
  - Change in model prediction Δz can be approximated by a linear function of the parameter changes Δp
  - Δp is (log-)normally distributed

$$\mathbf{C}_{\hat{z}\hat{z}} = \mathbf{J}\mathbf{C}_{pp}\mathbf{J}^T$$

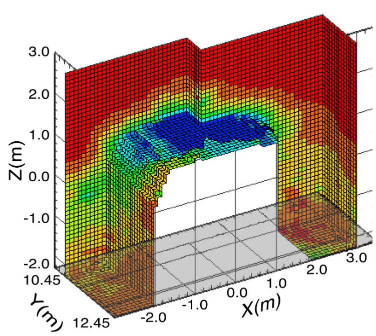
- Error band is *symmetric*, representing (log-) normally distributed prediction errors
- May assign certain probability to *unphysical* system behavior

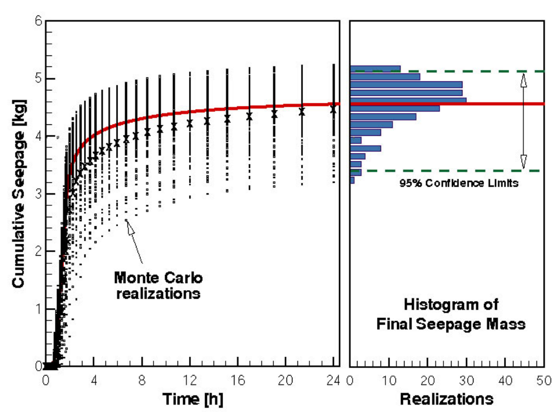
26

Monte Carlo Simulations

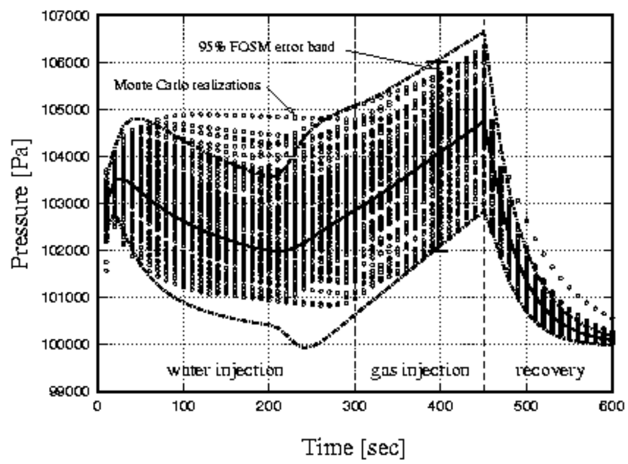
- Run many simulations with randomly selected parameter combinations drawn from the given probability density function.
- Provides *full distribution* of prediction uncertainty (histogram), which can be analyzed statistically.
- *Nonlinearities* are automatically taken into account.
- Results are always *physically reasonable*.
- Experimental designs: *Latin Hypercube Sampling.*
- Parameter *correlations* may be included.

Monte Carlo Simulations





Comparison FOSM-MC

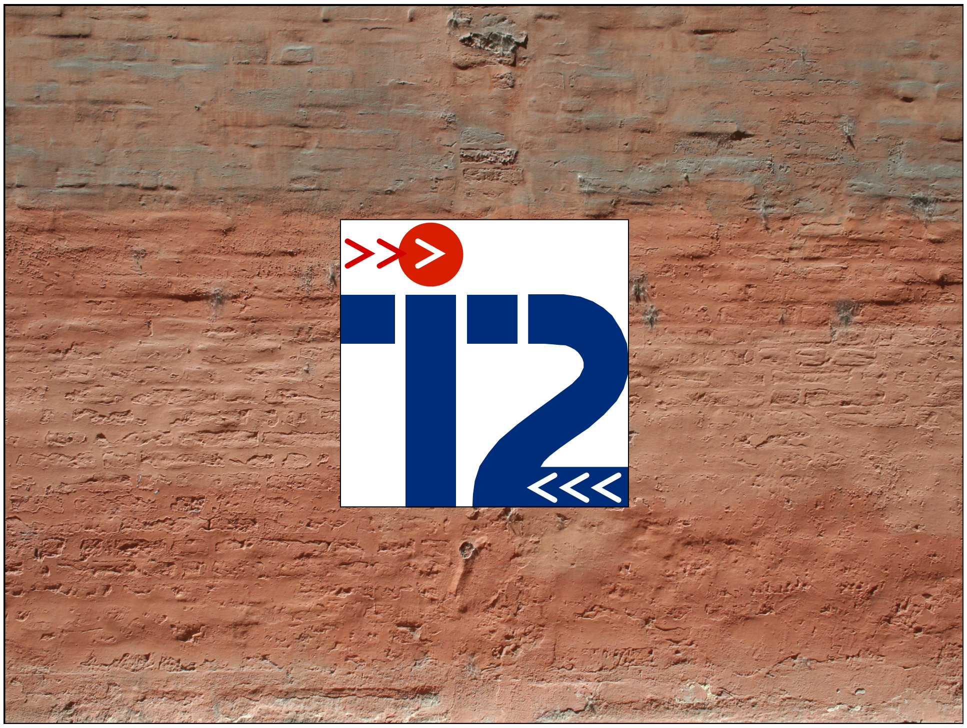


29

Uncertainty Analysis: Questions

6. Describe the main differences between FOSM and Monte Carlo simulations.
7. Assume you have to estimate the (hopefully small) probability that the TCE concentration at a drinking water well does not exceed a certain level.
   - Which uncertainty propagation analysis method would you choose?
   - Justify your choice.
   - Describe the procedure.

30



Uncertainty Propagation

Specify uncertainties in input parameters in block
`> PARAMETERS` or provide covariance matrix of estimated parameters using one of several formats

```
> COMPUTATION
  >> ERROR
     >>> FOSM
```

```
> COMPUTATION
  >> STOP
     >>> number of MC SIMULATIONS: 1000
     <<<
  >> ERROR
     >>> MONTE CARLO SEED: 9999 GENERATE
     >>> LATIN HYPERCUBE SAMPLING
     >>> EMPIRICAL ORTHOGONAL FUNCTION
```

32

Short Course

Solving Simulation-Optimization Problems Using iTOUGH2

Installation Instructions

The iTOUGH2 Flash Drive contains the following directories with files needed for the iTOUGH2 short course:

- Executables: iTOUGH2 executables for PC and Mac and related script files
- Exercises: Input files for computer exercises
- Lectures: Handout material in PDF format
- Manuals: TOUGH2 and iTOUGH2 manuals in PDF format

Installation on Mac

- Copy the contents of the iTOUGH2 flash drive to your hard drive's home directory ( ), maintaining the directory structure.
- Add directory that contains itough2 shell script file to command search path:
  - Open a Terminal ( )
  - Edit file *.bashrc* and add the following line:
    `PATH=$PATH:$HOME/iTOUGH2/Executables`
  - Save file and quit editor
  - Type `source .bashrc`
- Run iTOUGH2
  - Open a Terminal ( )
  - Change to the directory with the sample problems, e.g.:
    `cd ~/iTOUGH2/Exercises/Darcy`
  - Type `itough2` to see command usage and command line arguments
  - Run iTOUGH2 by typing:
    `itough2 darcy1i darcy 3 &`
- Edit and view input and output files
  - Use any text editor (vi, emacs, TextEdit, …) to edit and view iTOUGH2 input and output files. Use font `Courier` and a screen width of at least 132 columns.

Installation on PC

- Copy the contents of the iTOUGH2 flash drive to your hard drive, maintaining the directory structure.
- There are two ways to install and run iTOUGH2:
    - Option 1:
        - Copy the relevant executable (e.g., `it2_3.exe` for EOS3) from directory `…\iTOUGH2\Executables` to the working directory where all the input files are located (e.g., `…\iTOUGH2\Exercises\Darcy`)
        - Double-click on the executable
        - Enter the iTOUGH2 input file name (e.g. `darcy1i`)
        - Enter the TOUGH2 input file name (e.g. `darcy`)
    - Option 2:
        - Add directory that contains the `itough2.bat` batch file to command search path:
            - Locate path to directory *Executable*, typically: `C:\iTOUGH2\Execuatbles`
            - Open *START*, *Control Panel*, *System*
            - Open tab *Advanced*, click on *Environment Variables*
            - Under *System variables*, scroll to *PATH*, select it and click on *Edit*
            - Go to the end of the line and append a semicolon "`;`" followed by the full path to the directory *iTOUGH2\Executable*; click *OK*
        - Open a DOS-PROMPT window
            - *START*, *Run…*
            - Enter `cmd`
        - Run iTOUGH2
            - Change to the drive where you installed iTOUGH2 (e.g., type `C:`)
            - Change to the directory with the sample problems, e.g.: `cd C:\iTOUGH2\Exercises\Darcy`
            - Run iTOUGH2 by typing: `itough2 darcy1i darcy 3`
- Edit and view input and output files
    - Use any text editor (edit, Notepad, TextPad, WordPad, …) to edit and view iTOUGH2 input and output files. Use font `Courier` and a screen width of at least 132 columns.

Short Course

Solving Simulation-Optimization Problems Using iTOUGH2

iTOUGH2 Command Index

see also

Finsterle S., *iTOUGH2 Command Reference*,
Report LBNL-40041, Lawrence Berkeley National Laboratory, Berkeley California, 1999.

or

http://esd.lbl.gov/iTOUGH2/Command/command.html

GENERAL

```
> ECHO ON/OFF
HELP
INCLUDE FILE: file_name
#
/*
*/
```

PARAMETERS

```
> PARAMETER

  >> ABSOLUTE PERMEABILITY
  >> BIOT
  >> BOTTOMHOLE PRESSURE
  >> BOX-COX
  >> BULK DENSITY
  >> CAPACITY
  >> CAPILLARY PRESSURE FUNCTION
  >> COMPRESSIBILITY
  >> CONDUCTIVITY (WET/DRY)
  >> DILATION
  >> DRIFT
  >> ENTHALPY
  >> EXTERNAL
```

```
>> FACTOR
>> FRICTION ANGLE
>> FRICTION CORRECTION FACTOR
>> FRICTION FACTOR
>> FORCHHEIMER
>> GEOT
>> GUESS (FILE: file_name)
>> HARDENING
>> IFS
>> INITIAL (PRESSURE/: ipv)
>> KLINKENBERG
>> KURTOSIS
>> LAG
>> LIST
>> MINC
>> PARALLEL PLATE
>> PARMULT
>> PARSHIFT
>> PEST
>> POISSON
>> POROSITY
>> PRODUCTIVITY INDEX
>> PUMPING RATIO
>> RATE
>> REFLECTION
>> REGION (SINK/SOURCE, PERMEABILITY, OBSERVATION)
>> REGRESSION
>> REGULARIZATION (FILE: file_name) (BETA: beta)
>> RELATIVE PERMEABILITY FUNCTION
>> SCALE
>> SELEC
>> SHEAR
>> SHIFT
>> SKEWNESS
>> SKIN
>> STRAIN
>> TIME
>> TORTUOSITY
>> USER (: anno)
>> VOID FRACTION
>> YIELD
>> YOUNG
```

```
>>> DEFAULT
>>> LIST
>>> MATERIAL: mat_name (mat_name_i...) (+ iplus)
>>> MODEL
>>> NONE
>>> ROCK: mat_name (mat_name_i...) (+ iplus)
>>> SET: iset
>>> SINK: sink_name (sink_name_i ...) (+ iplus)
>>> SOURCE: source_name (source_name_i ...) (+ iplus)
    >>>> ANNOTATION: anno
    >>>> BOUND: lower  upper
    >>>> DEVIATION: sigma
    >>>> FACTOR
    >>>> GAUSS
    >>>> GUESS: guess
    >>>> INACTIVE
    >>>> INDEX: index (index_i ...)
    >>>> LOGARITHM
    >>>> LOG(F)
    >>>> NORMAL
    >>>> PARAMETER: index (index_i ...)
    >>>> PERTURB: (-)alpha (%)
    >>>> PRIOR: prior_info
    >>>> RANGE: lower  upper
    >>>> STEP: max_step
    >>>> UNIFORM
    >>>> VALUE
    >>>> VARIANCE: sigma^2
    >>>> VARIATION: sigma
    >>>> WEIGHT: 1/sigma
```

OBSERVATIONS

```
> OBSERVATION
  >> CONCENTRATION (comp_name/COMPONENT: icomp)
                   (phase_name/PHASE: iphase) (CHANGE/DELTA)
  >> CONTENT (phase_name/PHASE: iphase) (CHANGE/DELTA)
  >> COVARIANCE (FILE: filename)
  >> CUMULATIVE (comp_name/COMPONENT: icomp)
                (phase_name/PHASE: iphase) (CHANGE/DELTA)
  >> DRAWDOWN (phase_name/PHASE: iphase)
  >> ENTHALPY (phase_name/PHASE: iphase) (CHANGE/DELTA) (WELLHEAD)
  >> FLOW (phase_name/PHASE: iphase)
          (component_name/COMPONENT: icomponent) (HEAT)
          (CHANGE/DELTA)
  >> GENERATION (comp_name/COMPONENT: icomp)
                (phase_name/PHASE: iphase) (CHANGE/DELTA)
  >> HUMIDITY (CHANGE/DELTA)
  >> MASS FRACTION (comp_name/COMPONENT: icomp)
                   (phase_name/PHASE: iphase) (CHANGE/DELTA)
  >> MOLE FRACTION (comp_name/COMPONENT: icomp)
                   (phase_name/PHASE: iphase) (CHANGE/DELTA)
  >> MOMENT (FIRST/SECOND) (X/Y/Z) (CHANGE/DELTA)
       (comp_name/COMPONENT:icomp) (phase_name/PHASE: iphase)
  >> PEST (CHANGE/DELTA)
  >> PRESSURE (CAPILLARY) (CHANGE/DELTA) (WELLHEAD/BOTTOMHOLE)
              (phase_name/PHASE: iphase)
  >> PRODUCTION (comp_name/COMPONENT: icomp)
                (phase_name/PHASE: iphase) (CHANGE/DELTA)
  >> REGULARIZATION (FILE: file_name) (BETA: beta) (CHANGE/DELTA)
  >> RESTART TIME: ntime (time_unit) (NEW)
  >> SATURATION (phase_name/PHASE: iphase) (CHANGE/DELTA)
  >> SECONDARY (phase_name/PHASE: iphase) (: ipar) (CHANGE/DELTA)
  >> STEAM QUALITY (CHANGE/DELTA)
  >> TEMPERATURE (CHANGE/DELTA) (WELLHEAD)
  >> TIME: ntime (EQUAL/LOGARITHMIC) (time_unit)
  >> TIMES from DATA/OBSERVATIONS
  >> TOTAL MASS (comp_name/COMPONENT: icomp)
                (phase_name/PHASE: iphase) (CHANGE/DELTA)
  >> USER (: anno) (CHANGE/DELTA)
  >> VOLUME (phase_name/PHASE: iphase) (CHANGE/DELTA)
  >> WATERTABLE (CHANGE/DELTA)
```

\>>> CONNECTION: *elem1 elem2* (*elem_i elem_j ...*)
(++/+-/-+ *iplus*)
\>>> CONNECTION COORDINATES (BOX/ELLIPSOID/CYLINDER/CUBE) (ROTATE):
X1 Y1 Z1 (X2 Y2 Z2 (R)/(AZIMUTH DIP PLUNGE))
\>>> CONNECTION PROFILE/CROSS-SECTION/MAP
\>>> ELEMENT: *elem* (*elem_i ...*) (+ *iplus*)
\>>> ELEMENT COORDINATES (BOX/ELLIPSOID/CYLINDER/CUBE) (ROTATE):
X1 Y1 Z1 (X2 Y2 Z2 (R)/(AZIMUTH DIP PLUNGE))
\>>> ELEMENT PROFILE/CROSS-SECTION/MAP
\>>> MODEL
\>>> NONE
\>>> SINK: *sink_name* (*sink_namei* ...) (+ *iplus*)
\>>> SOURCE: *source_name* (*source_namei* ...) (+ *iplus*)

\>>>> ABSOLUTE
\>>>> ANNOTATION: *anno*
\>>>> AUTO (ADD NOISE)
\>>>> AVERAGE (VOLUME)
\>>>> BOX-COX: *lambda*
\>>>> COLUMN: *itime idata* (*istd_dev*)
\>>>> COMPONENT *comp_name*/: *icomp*
\>>>> DATA (*time_unit*) (FILE: *file_name*)
\>>>> DEVIATION: *sigma* (ADD NOISE)
\>>>> FACTOR: *factor*
\>>>> FORMAT: *format*
\>>>> HEADER: *nskip*
\>>>> INDEX: *index* (*index_i* ...)
\>>>> KURTOSIS
\>>>> LOGARITHM
\>>>> MEAN (VOLUME)
\>>>> PARAMETER: *index* (*index_i* ...)
\>>>> PHASE *phase_name*/: *iphase*
\>>>> PICK: *npick*
\>>>> POLYNOM: *idegree* (*time_unit*)
\>>>> REGRESSION: *rho*
\>>>> RELATIVE: *rel_err* (%) (ADD NOISE)
\>>>> SET: *iset*
\>>>> SHIFT: *shift* (TIME (*time_unit*))
\>>>> SKEWNESS
\>>>> SKIP: *nskip*
\>>>> SUM
\>>>> USER
\>>>> VARIANCE: *sigma^2* (ADD NOISE)
\>>>> WEIGHT: *1/sigma* (ADD NOISE)
\>>>> WINDOW (INDIVIDUAL/: *time_A time_B* (*time_unit*))

COMPUTATION

```
> COMPUTATION

  >> CONVERGE/STOP/TOLERANCE

     >>> ADJUST
     >>> ABORT (NO)
     >>> CONSECUTIVE: max_iter1
     >>> DELTFACT: deltfact
     >>> DIRECT
     >>> FORWARD
     >>> INCOMPLETE: max_incomplete
     >>> INPUT
     >>> ITERATION: max_iter
     >>> LEVENBERG: lambda
     >>> LIST
     >>> MARQUARDT: nue
     >>> REDUCTION: max_red
     >>> SIGNAL
     >>> SIMULATION: mtough2
     >>> STEP: max_step
     >>> UPHILL: max_uphill
     >>> WARNING

  >> ERROR
     >>> ALPHA: alpha (%)
     >>> EMPIRICAL (MATRIX: ndim (iTOUGH2)) (CORRELATION)
     >>> FISHER
     >>> FOSM (MATRIX: ndim (iTOUGH2)) (CORRELATION) (DIAGONAL)
     >>> HESSIAN
     >>> LATIN HYPERCUBE (CORRELATION/COVARIANCE) (DIAGONAL)
                         (MATRIX: ndim (iTOUGH2))
     >>> LINEARITY (: alpha (%))
     >>> LIST
     >>> MONTE CARLO (SEED: iseed) (GENERATE) (CLASS: nclass)
     >>> POSTERIORI
     >>> PRIORI
     >>> TAU: (-)niter

  >> JACOBIAN
     >>> CENTER
     >>> FORWARD (: iswitch)
     >>> HESSIAN
     >>> LIST
     >>> PERTURB: (-)perturb (%)
```

```
>> OPTION
    >>> ANDREWS: c
    >>> ANNEAL
        >>>> ITERATION: max_iter
        >>>> SCHEDULE: beta
        >>>> STEP: max_step
        >>>> TEMPERATURE: (-)temp0

    >>> CAUCHY
    >>> DESIGN
    >>> DIRECT
    >>> FORWARD
    >>> GAUSS-NEWTON
    >>> GRID SEARCH (: ninval1 (ninval2 (inval3)) /
                    FILE: filename) (UNSORTED)
    >>> L1-ESTIMATOR
    >>> LEAST-SQUARE
    >>> LEVENBERG-MARQUARDT (IDENTITY/EIGENVALUE)
                            (SUPER/TRUNCATED (: (-)truncation))
    >>> OBJECTIVE (: ninval1 (ninval2 (inval3))
                  FILE: filename) (UNSORTED)
    >>> PARALLEL: ncores (JACOBIAN / LEVENBERG: ncoreslm)
                  (SLEEP: isleep)
    >>> PEST
        >>>> DECPOINT: POINT/NOPOINT
        >>>> EXECUTABLE: file (BEFORE/AFTER)
        >>>> INSTRUCTION: num_instruction_files
        >>>> PRECISION: SINGLE/DOUBLE
        >>>> TEMPLATE: num_template_files
    >>> PVM: nhosts (JACOBIAN / LEVENBERG: nprocslm)
             (SLEEP: isleep) (FILE: node-file)
    >>> QUADRATIC-LINEAR: c
    >>> SELECT/SUPER
        >>>> CORRELATION: (-)rcorr
        >>>> IMMOBILIZATION (: ofredmin)
        >>>> ITERATION: niter
        >>>> SENSITIVITY: (-)rsens
        >>>> TRUNCATE (: (-)truncation)

    >>> SENSITIVITY
    >>> SEP (KURTOSIS: sepkurt) (SKEWNESS: sepskew)
```